Data-driven Recovery of Hand Depth using Conditional Regressive Random Forest on Stereo Images

ISSN 1751-8644
doi: 0000000000
www.ietdl.org

Rilwan Remilekun Basaru[1]*, *Chris Child*[1], *Eduardo Alonso*[1], *Gregory Slabaugh*[2]

[1] *Department of Computer Science, City University London, Northampton Square, London, United Kingdom*
[2] *Huawei Technologies Research & Development (UK) Ltd*
* *E-mail: Remilekun.basaru.1@city.ac.uk*

Abstract: Hand pose is emerging as an important interface for human-computer interaction. This paper presents a data-driven method to estimate a high-quality depth map of a hand from a stereoscopic camera input by introducing a novel superpixel-based regression framework that takes advantage of the smoothness of the depth surface of the hand. To this end, we introduce Conditional Regressive Random Forest (CRRF), a method that combines a Conditional Random Field (CRF) and a Regressive Random Forest (RRF) to model the mapping from a stereo RGB image pair to a depth image. The RRF provides a unary term that adaptively selects different stereo-matching measures as it implicitly determines matching pixels in a coarse-to-fine manner. While the RRF makes depth prediction for each super-pixel independently, the CRF unifies the prediction of depth by modeling pair-wise interactions between adjacent superpixels. Experimental results show that CRRF can generate a depth image more accurately than the leading contemporary techniques using an inexpensive stereo camera.

1 Introduction

Recently there has been a surge in interest in virtual reality devices, augmented reality devices, and other egocentric devices including handheld and wearable smart cameras [1]. Such devices typically do not have a keyboard or mouse interface and therefore require new modes of human-computer interaction. The research in this paper is motivated by advances in human body pose estimation [11, 14, 28] that have enabled new applications in gesture control, computer games, person tracking, action recognition and action tracking. While human body pose estimation from RGBD data may be considered a solved problem [11, 14, 28], open challenges remain for estimating hand pose [21, 22], as hands exhibit a high degree of self-occlusion and greater variation in orientation relative to the camera. We argue that the key to natural gestural interaction with next-generation devices is robust hand pose estimation. Indeed, hands have attracted much research interest in recent years and received special focus in workshops at leading computer vision conferences.

An important design criterion for a hand pose estimation approach is the type of imaging sensor employed. RGBD sensors are a popular choice, as depth-based input provides good shape information, robustness to clutter and changes in ambient conditions. Using the depth channel, inference algorithms can be developed to estimate the hand pose. Despite the successes of such approaches, depth channel data capture poses several limitations, including poor form factor in egocentric applications, large energy consumption, poor near distance coverage, and inferior performance outdoors. These deficiencies may render such devices impractical for egocentric scenarios that the aforementioned devices bring. Therefore, in this paper, we focus instead on RGB data capture. By acknowledging that a single RGB camera does not provide enough shape and structure information, we focus on a stereovision technique using two cameras.

As discussed in the introduction active RGBD sensors have become the more prominent approach to the problem of inferring hand articulation [10, 11, 21, 22] as opposed to RGB monocular sensors. There are fewer methods in the literature that attempt to resolve for pose from stereo capture. Most pose recovery frameworks from depth can be differentiated based on the model type, which is either discriminative or generative. With generative model-based technique, a hypothesis of visual data of the hand is generated using computer graphics, often using an articulated rendered 3D hand model. A very prominent framework used in generative models is Particle Swarm Optimization (PSO) as in [9, 10, 57]. In these approaches, hand pose estimation and tracking are demonstrated in hand-object and hand-hand interacting scenarios by optimizing for the parameters that yield the rendered depth that best matches the observed depth. [10] jointly solves for pose and shape estimation by skinning rigged models. Unlike [10], [11] avoids the need of graphically synthesized hand model by generating gaussian based on the hand articulation at a tracking instance. To evaluate the correspondence of a proposed articulation during a tracking instance, the correlation between the sum-of-gaussians generated based on joint location (in the proposed articulation) and that generated based on a point cloud estimation of the observed depth image is used. Discriminative models, on the other hand, constitute probability distribution of the pose of the hand whose parameters are dependent on the observed depth image. A typical example of this is [54, 68]. These techniques aim to first establish the spatial position of each joint of the hand by classifying each pixel before computing the general pose of the hand.

The goal of our research is to extract robust hand depth information from stereo RGB inputs as a precursor to hand pose estimation. Human vision, which can efficiently discern articulations and perform tracking activities, employs stereo imaging. Depth estimation from two views has a long and rich history in computer vision and fundamentally relates to establishing correct correspondences between images. However, the recovery of hand depth provides unique challenges that differentiate the problem from depth recovery of arbitrary scenes as expressed in [5]. Unlike generic scene depth estimation there is significantly less texture, which makes stereo matching substantially more challenging. There is also a high tendency of self-occlusion which manifests in changes in depth that might not reflect in a change in texture. For example, the occlusion of a finger on the palm will yield a change in depth but the color and the texture of the region of occlusion might remain unchanged as the color of the skin might be consistent (whether on the finger or on the palm region). This necessitates a new hand-specific depth estimation technique to outperform generic stereo matching algorithms.

Whilst recovery of hand depth provides challenges as previously expressed, the constraint that the depth recovery task will only apply to a particular class of object (hand) means that stereo matching constraints can be learned using a machine learning approach and tested on similar surfaces. This is particularly useful as we can better establish the matching criteria that can achieve the best stereo matches and hence disparity since we know the typical structure of the "scene" for which we are going to be estimating depth. Specifically, in this paper, using inexpensive stereo imaging setup as shown in Fig. 1, we recover accurate depth images of hand poses that can be used as a pre-step to gesture recognition. In this work, we do not implement gesture recognition, instead, we solely focus on recovering accurate depth. The proposed technique also relies on a robust hand segmentation procedure. We do not address this in this paper as there is a large body of literature on this subject (see, for example, [6, 7, 8]).

Conventional approaches to stereo-matching rely on universal conditions for finding correspondences. Specifically, to our problem of hand-based stereo-matching, we propose a more robust approach that adaptively establishes matching conditions based on unique properties of the hand (e.g., skin tone, texture, etc.). Underlying our approach are four main conjectures. The first is that the depth surface of a scene with hand poses consists of a set of homogenous regions that yield a smooth surface and continuous texture. Second, that when establishing correspondence in untextured regions, a higher number of variants of the matching cost (i.e., type of cost function, size of window, etc.) improves the chances of discerning between ambiguous matches. Using different matching criteria to assess potential matches effectively increases the dimension of the feature space that is used to determine similarity. This is particularly the case with attempting to establish correspondence on an inherently untextured hand region. Third, that the difference in skin tone and hand size for different individuals makes establishing universal matching criteria for determining stereo correspondence a difficult task. Conventional approaches to stereo-matching adopt this universal approach when attempting to establish a single cost function for appraising the similarity of potentially matching correspondences. In our work, we propose that a more robust approach will be to adaptively establish matching conditions based on specific properties of the hand (e.g. skin tone etc.). Last, that the most effective approach to stereo correspondence search is a coarse-to-fine one, which we implicitly manifest in a machine learning context.

1.1 Our Contribution

This paper proposes a novel, data-driven Regressive Random Forest framework that learns the mapping between a stereo image pair and high-quality ground truth depth measurement. In so doing, we present the Conditional Regressive Random Forest (CRRF), an innovative combination of Regressive Random Forest and Conditional Random Fields to model this mapping. The paper makes the following contributions:

1. We introduce a machine learning approach to establish stereo correspondences, by solving a superpixel-based regression problem rather than explicitly minimizing a stereo matching cost function.
2. Rather than rely on a single cost function or single window size, our method fuses multiple cost functions computed over different window sizes as input to the regressor. Our experimental results demonstrate that the regressor performs estimation in a coarse-to-fine estimation.
3. The regressor is trained using expert trees that learn from different subsets of the data, based on holistic hand features like skin tone.
4. Our CRRF framework combines a unary term (regression) and pair-wise term (smoothness). To solve the inference problem, we derive a closed-form non-iterative solution, unlike conventional CRF methods that require iterative solvers.
5. We demonstrate that CRRF outperforms other methods for depth estimation of a hand from stereo inputs, greatly outperforming generic algorithms for stereo matching.

In this paper, we apply CRRF to hand depth estimation. However, we note the framework is general and can be applied to other imaging problems that involve supervised learning (classification, regression) on superpixels. The rest of the paper is structured as follows: the next section presents a general survey of related work in the field of depth estimation and CRRF. Section 3 gives a brief overview of our methodology while in Section 4 we give a more detailed presentation of the Random Forest and CRF component of our technique. Section 5 elaborates on the details of our implementation of the proposed technique. Experiments and results are discussed in Section 6. The paper concludes in Section 6 with a review of our main contributions and a brief discussion of future work.

2 Related Work

Depth recovery from stereo matching is a passive technique based on the concept of stereopsis. A scene is captured from multiple perspectives and the displacement of each pixel point between each image is computed so that the distance of the actual world point to the camera is inversely proportional to the displacement in its corresponding image pixel. The Middlebury website [3] contains a large collection of algorithms and cost functions, as well as a test-bed for relative comparison. Another related area is depth recovery from a single image. [11], [18] and [19] model the depth estimation as a Markov Random Field (MRF) learning problem. The success of Deep Learning in computer vision has prompted recent approaches to model the problem with Convolutional Neural Networks (CNNs) [20]. While showing much promise, work to date has lacked stronger geometric features (like stereoscopic information) highly correlated with depth. A closely related technique to ours is [5], where a data-driven approach has been taken to develop a near-infrared based depth camera. In this study, a two-layered Random Forest framework was used to establish the mapping between near-infrared images of a scene consisting of articulated hand poses captured from modified RGB cameras to actual depth. While this is a unique and relatively inexpensive technique, it suffers from ambient infrared radiation (e.g., when used in an outdoor scene). In addition, it requires non-trivial hardware modifications.

Our work is also related to [28], where the prediction of joint locations that are prominently modeled with a Random Forest is conditioned on global variables (like torso orientation). A major difference is that we explicitly combine Random Forest and Conditional Random Fields. To the best of our knowledge, the closest

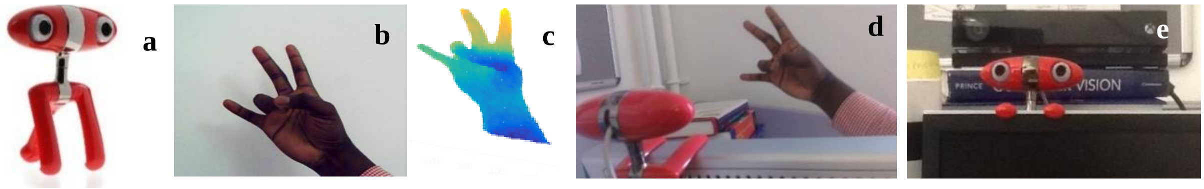

Fig. 1: Using an inexpensive stereo camera (a), RGB images of the hand from two perspectives are captured (b, c), and mapped to an accurate depth image (d). The proposed technique can potentially use a stereo rig system to estimate hand articulation and pose (e). Stereo camera and RGBD camera setup used for data capture.

approach in literature is [27], which attempts to solve the problem of multiclass object recognition and segmentation by modelling perceptual organization (e.g., surrounding pixels are correlated) and context-driven recognition (e.g., that establishing an object is in the scene may indicate that another object will be in the scene) using a CRF. CRF inference in [27] is achieved using the Swendsen-Wang cut algorithm that iterates Metropolis-Hastings jumps. These approaches differ from ours in that we adaptively combine prediction from the trees using the unary term of our CRRF whilst the pairwise term maintains spatial pixel depth constraints. Also, we present a closed form solution to inference on our Conditional Random Regressive Forest. This contrasts with earlier approaches like [27] that apply an iterative approach to achieving inference. There has been a recent increase in interest in hand pose estimation, with several techniques proposed, particularly those working with data captured from active depth sensors or monocular cameras [11], [21] and [22]. However, less work has been done on hand pose recovery based on stereoscopic images [4] and [23]. [23] uses recovered disparity information from stereo data, to determine the arm orientation and the hand location; and in turn, initialize color-based segmentation of the hand. It uses the recovered arm orientation to achieve perspective unwarping of the hand into "easily recognizable gesture template" [23]. [4] presents a real-time recognition of hand gestures from stereo inputs. It applies a rule-based approach to combine information from stereo pairs of hand captures to improve hand detection before performing gesture recognition. In effect, the stereo information is used to establish a more robust contour of the hand. Note that this does not explicitly establish a depth map it simply uses the stereo information to recover the hand contour. Hence it still does not provide shape information (present in the depth map) that improves the robustness of gesture recognition. We contribute to this area by developing a machine learning framework that recovers depth from stereo.

3 Overview of Conditional Regressive Random Forest (CRRF)

Our method recovers a high-quality depth image from two stereoscopically acquired images of the hand. Our dataset captures the hand in a variety of poses. Fig. 2 shows an overview of the approach. First, we segment the reference stereo image into superpixels using SLIC [9]. For every superpixel that lies within the hand region, we compute its stereo matching cost with all potentially matching pixels along the Epipolar line in the corresponding image. We apply five different matching cost functions simultaneously. Each of the stereo matching cost functions is applied under varying window sizes that are centered on the centroid of the superpixel, and on the potentially matching pixels in the corresponding stereo pair. The matching cost values that are computed across all combinations of cost functions, window sizes and potentially matching pixels are concatenated to a single feature vector. Henceforth we will refer to this vector of features as the matching-cost feature vector. Note that we do not attempt to identify matching pixels explicitly; we simply compute the matching-cost feature vector (for each superpixel). In addition, we extract features that relate to the hand in the scene. These features primarily represent how far away the entire hand is from the camera, texture, and the color of the skin. We refer to this as the holistic hand feature vector.

A Regressive Random Forest (RRF) is trained to regress for the depth of a superpixel based solely on its matching-cost feature, however, each tree in the RRF is exposed to a subset of the training data based on its holistic hand feature. Finally, we use a CRF framework

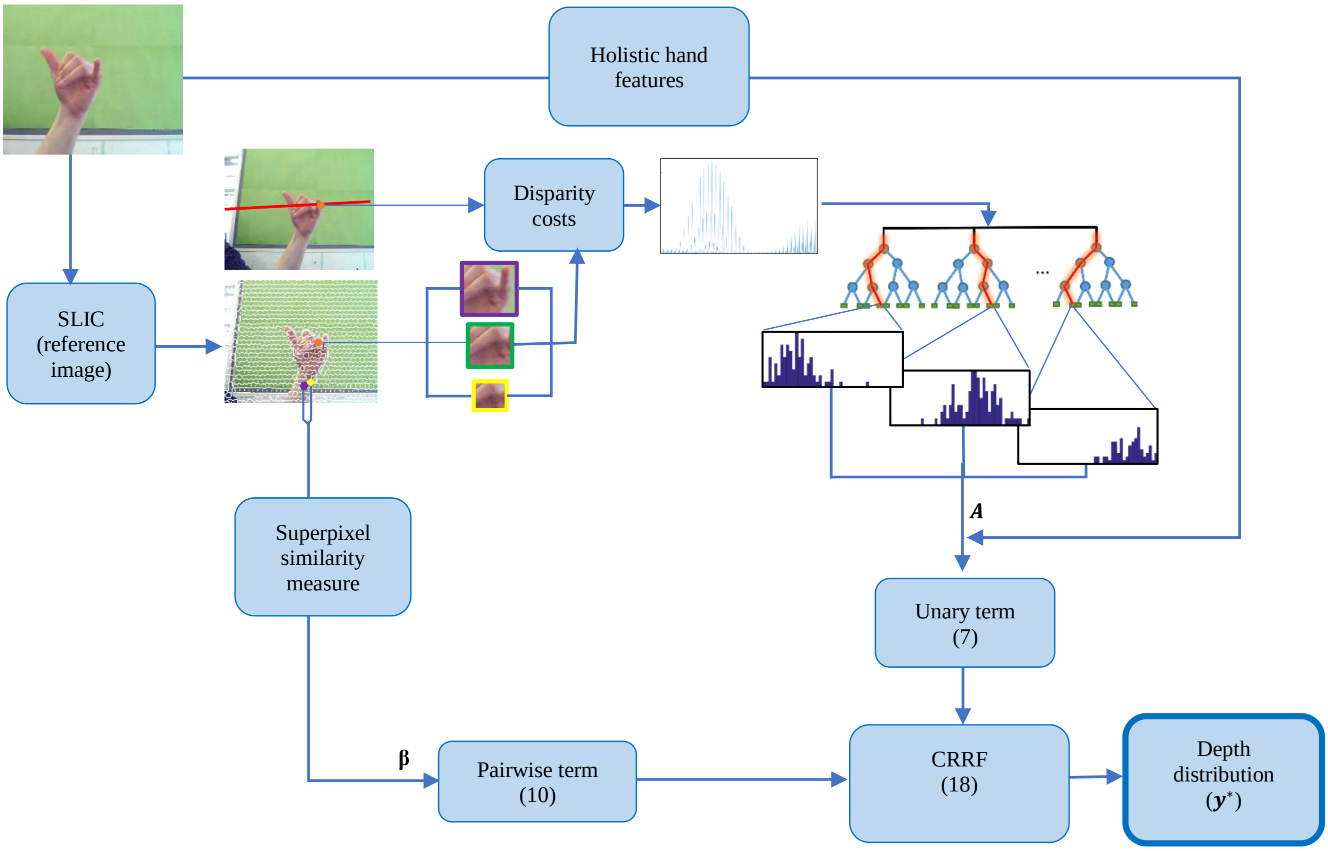

Fig. 2: An illustration of the proposed approach. First, the reference stereo image is segmented into superpixels. Using different window sizes and cost functions, the disparity cost along the Epipolar line in the corresponding image is computed. This cost is concatenated to generate a feature signal that is fed into a Regressive Random Forest. Posterior probability distributions from the trees are combined using the matrix, $\boldsymbol{A}$ (used to compute the unary term of the CRRF model). The similarity measure between neighboring superpixels is multiplied with $\boldsymbol{\beta}$ to yield the pairwise term. The CRRF is solved in a closed form solution, $\boldsymbol{y}^*$, that maximises (11)

to combine the predictions from each tree in the RRF whilst constraining for smooth depth surface prediction. We delve into greater detail of above overview in the following section.

3.1 Notation

For ease of presentation, vectors and matrices are denoted with a boldface-lowercase and boldface-uppercase respectively. Vector and matrix transpose are denoted with an upper script T, as in $\{\}^T$. Also that all vectors are assumed to be column vectors e.g. $p = [p_x, p_y, p_z]^T$. A vector with one element only is denoted as $\boldsymbol{i}$, whilst $\boldsymbol{I}$ denotes the identity matrix.

For a given reference image, z, and its corresponding stereo image, z', a hand superpixel in z is denoted as $x_j \in \{x_1, ..., x_J\}$ and the centroid pixel of the superpixel as $\boldsymbol{v}_j$. For each $\boldsymbol{v}_j$, we establish a search space of W potentially matching pixels, $\boldsymbol{v}_{j,w} \in \{\boldsymbol{v}'_{j,1}, ..., \boldsymbol{v}'_{j,W}\}$ located in z'. Observe Fig. 2, and note that only one of the stereo image pair (the reference image) is segmented (using SLIC) into superpixels, hence the search space $\boldsymbol{v}_{j,w}$, is a contiguous subset of all pixel points along the corresponding Epipolar line of $\boldsymbol{v}_j$. Whilst it would be possible to run SLIC on the other image in the stereo pair, and try to match superpixels, this would be problematic as there is no guarantee that the superpixels in the other image would have a similar size and distribution to the ones generated in the reference image. To avoid this problem, we perform matching at the pixel level. The vector

$$\boldsymbol{c}_{k,g}(\boldsymbol{v}_j) = \quad . \quad [f_{k,g}(\boldsymbol{v}_j, \boldsymbol{v}'_{j,1}), f_{k,g}(\boldsymbol{v}_j, \boldsymbol{v}'_{j,2}), ..., f_{k,g}(\boldsymbol{v}_j, \boldsymbol{v}'_{j,W})], \tag{1}$$

where $f_{k,g}$ is the resulting cost from using the k^{th} matching cost function, and g^{th} window size. $\boldsymbol{c}_{k,g}(\boldsymbol{v}_j)$ is concatenated for all combinations of k and g to get a single matching-cost feature vector. Hence for each superpixel, x_j, given that $k \in \{1, ..K\}$ and $g \in \{1, ..G\}$, the corresponding matching-cost feature will be $\boldsymbol{c}_j \in \mathbb{R}^N$ where $N = W * G * K$. Note that W, G and K are the number of pixels in the search space, the number of window sizes, and the number of matching cost functions respectively. Also observer how $\boldsymbol{c}_{k,g}(\boldsymbol{v}_j)$ is a vector of matching cost values between the pixel of interest, $\boldsymbol{v}_j$ (in the reference stereo image) and each potentially matching pixel in the corresponding stereo image under a matching cost function, k, and a window size, g. The groundtruth depth at the centroid pixel,$\boldsymbol{v}_j$, is d_j, the regression dataset is then defined as $\{(d_1, \boldsymbol{c}_1)^{(z)}, ..., (d_J, \boldsymbol{c}_J)^{(z)}\}$ for all Z stereo image pairs collected over different hand poses and subjects. These extracted feature is fed into our Random forest based framework. We give describe our Random forest framework in the succeeding subsections.

3.2 Expert Random Forest

N decision trees are grown by recursively splitting and passing training data, S, into two sub nodes Si. The splitting is based on features. Following [29], randomness is maintained in feature selection and threshold selection as the tree aims to decrease the entropy of the training dataset by maximizing the Information Gain,

$$I_G(\theta) = E(S) - \sum_{i \in \{L,R\}} \frac{|S_i(\theta)|}{|S|} E(S_i(\theta)). \tag{2}$$

Entropy is defined as

$$E(S) = \log(\sigma_s), \tag{3}$$

where σ_s is the standard deviation of the depth values of the centroid pixel points within the subset, S. The intuition is that the trees implicitly learn how to adaptively select the size of window and type of cost function based on different tree split levels. This is analogous to adaptively determining the size of the window and type of cost function to use at different stages of a coarse-to-fine approach to searching for pixel correspondence. The entropy decreases moving through each tree from the root node to the leaf nodes. Experimental results will show that the entropy is related to the coarse-to-fine selection of features.

Expert Trees: As previously stated, holistic hand features (features that describe the entire hand), are additionally computed. This step is motivated by the significant effect that skin color and the overall distance of the hand have on the matching-cost features. Consequently, establishing a stereo-matching criterion (i.e., matching cost, window size, etc.) that works effectively across different skin tones and hand depth levels is a difficult task. To this end, all the stereo image pairs are clustered into classes based on their holistic hand features. Each tree in the RRF is trained by bagging from only one of the classes, making it an expert at regressing the depth for that class. Thus, a particular tree may be expert at predicting the depth of superpixels in a darker-toned hand that is closer to the camera, whilst another may specialize in lighter-toned hands that are farther away. See Section 5.2 for more detail on holistic hand features. When predicting the depth of an unseen stereo pair with a holistic hand feature, the CRF framework, discussed in the next subsection, ensures that more emphasis is placed on prediction from expert trees with similar holistic hand features than to others. Now we have established our Random forest framework, in the next subsection we describe how we selectively bias for each class of trees using the holistic hand feature and account for continuous depth estimate using our CRRF framework.

3.3 CRRF Framework

This section describes the CRRF framework (using the same notation). Consider a new stereo image pair, with a holistic hand feature vector, $\boldsymbol{h}$, whose superpixels' depths are to be predicted using the trained RRF. For a single superpixel, x_j, each RRF tree, t, produces a posterior probability distribution, $p_t(d_j|\boldsymbol{c}_j)$. This distribution is discretized by quantizing the depth values into D finite values. This yields a probability vector, $\boldsymbol{p}_{t,j} \in \mathbb{R}^D$ that is then consolidated across all the T trees into $\boldsymbol{P}_j = [\boldsymbol{p}_{1,j}, \boldsymbol{p}_{2,j}, ..., \boldsymbol{p}_{T,j}] \in \mathbb{R}^{D \times T}$. The probability of d_j, given the reference stereo image and trained RRF, $Pr(d_j|\boldsymbol{P}_j, \boldsymbol{h})$, is modelled as a CRF. Conventionally a CRF formulates conditional probability as a product of potentials, that is

$$Pr(a|b) = \frac{1}{Z(b)} \prod_i \exp(\phi_i) = \frac{1}{Z(b)} \exp\left[\sum_i (\phi_i)\right], \tag{4}$$

where $Z(b)$ is the partitioning function, and ϕ_i are potentials [30]. Inspired by [15], the potentials in the proposed framework take the form of a unary E_U and a pairwise term E_P. The conditional probability is approximated because of the intractable nature of $Z(b)$ (as it requires an integral over all combination of all possible states that the target and input variable could have)in the proposed framework,

$$\widetilde{Pr}(\boldsymbol{d}_j|\boldsymbol{P}_j, \boldsymbol{h}) = \exp\left[\sum_c (\phi_c)\right] = \exp[E_U + E_P], \tag{5}$$

where $\widetilde{Pr}$ denotes an unnormalized probability distribution. This approximation will suffice because the objective is to estimate the depth level with the maximum probability. Hence, the probability of the depth distribution function for all superpixels given $\boldsymbol{P}_j$ and the image's holistic hand feature, $\boldsymbol{h}$, is represented as the exponent of sums of both potentials. While the unary term aims at yielding a conditional probability distribution that maximizes the probability of the true depth level, the pairwise term encourages neighboring superpixels to have a similar posterior probability distribution.

Unary Potential: The unary term predicts the depth level of a superpixel based on its posterior distribution from the RRF trees and the holistic hand feature. To this end a unary weighting matrix, $\boldsymbol{A} \in \mathbb{R}^{T \times H}$, is introduced, which weights the posterior from each tree based on $\boldsymbol{h} \in \mathbb{R}^H$. This is important because expert trees are

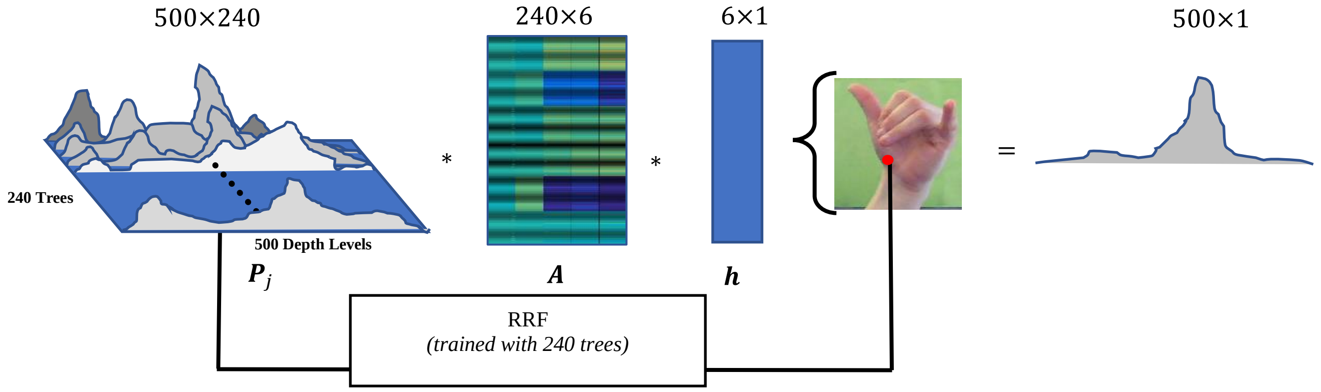

Fig. 3: An illustration of the unary potential when the number of trees, $T = 240$, the number of depth levels, $D = 500$ and the number of holistic hand features, $H = 6$. This illustrates how $\boldsymbol{A}$ weights the posterior probability, $\boldsymbol{P}_j$, from the trees using $\boldsymbol{h}$ to give a probability distribution of a single superpixel. This becomes the unary term in the CRRF.

trained, as opposed to randomly bagged trees. The $\boldsymbol{A}$ matrix provides weights to trees depending on the holistic hand feature. Hence it places varied emphasis on the predictions from different trees.

Taking inspiration of the Bhattacharyya metric [17], E_U is formulated as an affinity measure between true depth probability, $\widehat{\boldsymbol{p}}_j^T$, and the predicted probability, $\boldsymbol{P}_j\boldsymbol{Ah}$, as in,

$$E_U = \frac{1}{J}\sum_{j=1}^{J}\left[\frac{\widehat{\boldsymbol{p}}_j^T\boldsymbol{P}_j\boldsymbol{Ah}}{\boldsymbol{i}^T\boldsymbol{Ah}}\right]. \qquad (6)$$

This is accumulated across all superpixels in the reference stereo image. The denominator in (6) ensures that $\boldsymbol{P}_j\boldsymbol{Ah}$ remains normalized. The surface plot in Fig. 4 shows how the different entries of $\boldsymbol{A}$ vary relatively. Figs. 3 and 4 give an illustration of the weighting ability of $\boldsymbol{A}$. The peaks indicate a strong relationship between entries of $\boldsymbol{h}$ and the tree index. Studying both figures, consider a hypothetical example where $\boldsymbol{h} = [0, 0, 0, 1, 1, 1]^T$. In this case, the holistic hand feature vector will weight the prediction from the 240 trees based on the last three columns of $\boldsymbol{A}$, thereby giving less weighting to trees 40 to 80 and trees 160 to 200.

Let $\widehat{\boldsymbol{y}} = [\widehat{\boldsymbol{p}}_1^T, \widehat{\boldsymbol{p}}_2^T, ..., \widehat{\boldsymbol{p}}_j^T] \in \mathbb{R}^{(D*J)}$ be a vector resulting from the concatenation of the actual probability distribution of all hand region superpixels and let $Y = [\boldsymbol{P}_1, \boldsymbol{P}_2, ..., \boldsymbol{P}_J]^T \in \mathbb{R}^{(D*J)\times T}$ be the matrix whose row vectors are the concatenation of the predicted probability distribution from each tree. Then the unary potential in (6) can be rewritten for all superpixels in a single stereo image, z, in matrix form as follows:

$$E_U = \frac{1}{J\boldsymbol{i}^T\boldsymbol{Ah}}\widehat{\boldsymbol{y}}^T\boldsymbol{Ah}. \qquad (7)$$

The larger E_U becomes, the more similar the consolidated predicted probability, $\boldsymbol{P}_j\boldsymbol{Ah}$, is to the true depth probability, $\widehat{\boldsymbol{p}}_j^T$.

Pairwise Potential: The pairwise potential enforces the constraint that adjacent superpixels often possess similar depth and hence similar probability distributions. This is based on the smooth nature of the depth of the hand surface. Similar to [13], a visual similarity measure between neighborhood superpixels is established to apply an adaptive depth similarity constraint. Specifically, neighboring superpixels that appear dissimilar in terms of color, texture, and size will have a weaker pairwise potential encouraging similar predicted depth. This is particularly intuitive in a self-occluded scenario.

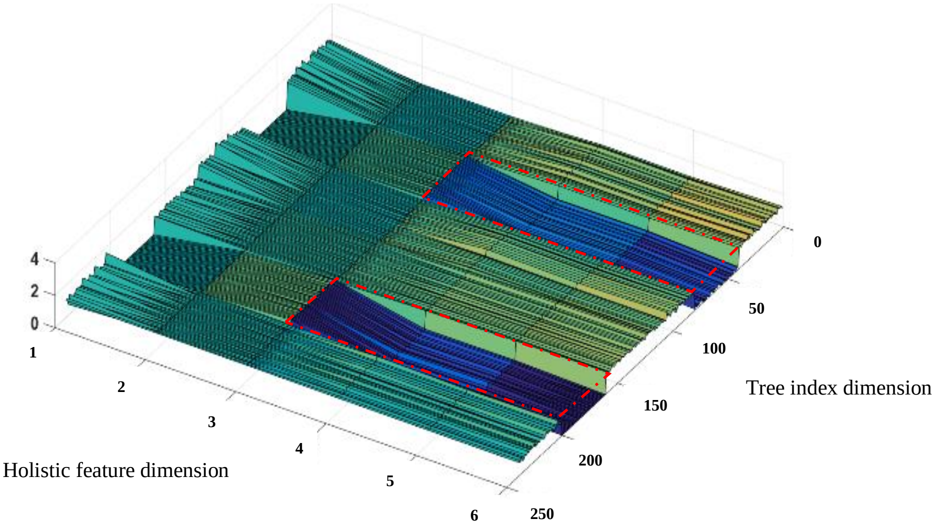

Fig. 4: A surface plot of the matrix $\boldsymbol{A}$ (see Fig 3, used to weigh the expert trees based on the holistic hand feature. A higher value indicates more weight. Consider a hypothetical holistic hand feature vector, $[0, 0, 0, 1, 1, 1]$, which, when post-multiplied with $\boldsymbol{A}$ will give less weighting to trees 40 to 80 and 160 to 200 based on their lower values (bluer colors), highlighted with red boxes.

Table 1 A Brief outline of key components of the proposed framework. This includes Matching-cost feature, Holistic Hand feature, Superpixel Similarity measure, Expert trees, Unary Term, Pairwise Term, and the CRRF formulation.

Components	**Implication**
Matching-cost features (per superpixel)	This feature vector describes how similar/dissimilar the centroid of the superpixel is to all pixels along the Epipolar line on the corresponding stereo image. This is potentially determined by the disparity at that centroid pixel.
Superpixel Similarity measure	This is a vector of metrics that conveys how similar or dissimilar two neighboring superpixels are.
Holistic Hand features	This feature vector describes the general shift, tone, and size of the hand.
Expert Trees	RRF are conventionally built, however, each tree is trained on a dataset of hands captures of a particular class, based on its Holistic hand feature.
Unary Term	During a superpixel depth prediction, the unary term facilitates the bias to predictions from expert trees that were trained from a dataset of similar Holistic hand features.
Pairwise Term	The pairwise term adds the constraint that superpixels depth predictions yield a continuous surface in a neighborhood, i.e. neighboring superpixels (particularly those with high Superpixel Similarity measure) will tend to have similar depth predictions.
CRRF Formulation	The CRRF formulation yields a closed form solution to superpixel depth prediction that combines the unary and pairwise terms.

The discontinuity in texture resulting from a finger occluding the palm, for example, will indicate that a lower smoothness constraint is placed on neighboring superpixels that exist on the edge of the finger and the palm.

To achieve this behaviour, a similarity vector, $s_{j,k} = [s^1_{j,k}, ..., s^Q_{j,k}]$, and a pairwise weighting, $\boldsymbol{\beta} \in \mathbb{R}^Q$, are introduced. For a pair of neighbouring superpixels, x_j and x_k, Q superpixel similarity measures are computed between them (more details on the superpixel similarity measures are presented in Section 5.2. Pairwise potential is specified as:

$$E_P = \frac{1}{|U|} \sum_{(j,k)\in U} \boldsymbol{\beta}^T \boldsymbol{s}_{j,k} \widehat{\boldsymbol{p}}_k^T \widehat{\boldsymbol{p}}_j \tag{8}$$

where U is a set of all possible pairs of neighbouring hand superpixels. Subsequently, the pairwise potential is a measure of the affinity of the probability of all pairs of neighbouring superpixels, and $\boldsymbol{\beta}^T \boldsymbol{s}_{j,k}$ determines the contribution of each pair of superpixels to this measure.

Let $\boldsymbol{B} \in \mathbb{R}^{J\times J}$ be a matrix such that, its elements are given by

$$\boldsymbol{B}_{j,k} = \boldsymbol{\beta}^T \boldsymbol{s}_{j,k} \boldsymbol{I}, \tag{9}$$

and zeros everywhere else. $\boldsymbol{I}$ is a $\boldsymbol{D} \times \boldsymbol{D}$ identity matrix. With this matrix, the pairwise potential in (8) can be represented in matrix form as:

$$E_P = \frac{1}{|U|} \widehat{\boldsymbol{y}}^T \boldsymbol{B} \widehat{\boldsymbol{y}}. \tag{10}$$

A resulting depth image with high level of smoothness will yield a large pairwise potential, E_P.

Complete CRRF: At this stage, both potentials, unary and pairwise, have been established and the higher they are, the smoother and the more likely the predicted depth becomes (based on its probability). (5), (6) and (8) are combined to result in

$$\widetilde{Pr}(\boldsymbol{d}_j|\boldsymbol{P}_j, \boldsymbol{h}) = \exp\left[\frac{1}{J}\sum_{j=1}^{J}\left[\frac{\widehat{\boldsymbol{p}}_j^T \boldsymbol{P}_j \boldsymbol{A}\boldsymbol{h}}{\boldsymbol{i}^T \boldsymbol{A}\boldsymbol{h}}\right] + \frac{1}{|U|}\sum_{(j,k)\in U} \boldsymbol{\beta}^T \boldsymbol{s}_{j,k} \widehat{\boldsymbol{p}}_k^T \widehat{\boldsymbol{p}}_j\right], \tag{11}$$

for a single stereo image pair. In this unified framework, the aim is to maximize (11) based on $\boldsymbol{A}$ and $\boldsymbol{\beta}$. For all stereo images in the training set, z, the framework attempts to maximize $\sum_z \log \widetilde{Pr}(\boldsymbol{y}^{(z)}|\boldsymbol{P}^{(z)})$. Formally,

$$\max_{\boldsymbol{A}\geq \boldsymbol{0},\boldsymbol{\beta}} \sum_{z=1}^{Z} \log \widetilde{Pr}(\boldsymbol{y}^{(z)}|\boldsymbol{P}^{(z)}) + \lambda(1 - \boldsymbol{\beta}^T\boldsymbol{\beta}) \tag{12}$$

where λ is the decay weight on the constraint with $\boldsymbol{\beta}$ maintaining a unit length and

$$\log \widetilde{Pr}(\boldsymbol{d}_j|\boldsymbol{P}_j, \boldsymbol{h}) = \frac{1}{J}\sum_{j=1}^{J}\left[\frac{\widehat{\boldsymbol{p}}_j^T \boldsymbol{P}_j \boldsymbol{A}\boldsymbol{h}}{\boldsymbol{i}^T \boldsymbol{A}\boldsymbol{h}}\right] + \frac{1}{|U|}\sum_{(j,k)\in U} \boldsymbol{\beta}^T \boldsymbol{s}_{j,k} \widehat{\boldsymbol{p}}_k^T \widehat{\boldsymbol{p}}_j. \tag{13}$$

The monotonic nature of log functions implies that (13) increases as $\boldsymbol{P}_j\boldsymbol{A}\boldsymbol{h}$ and $\widehat{\boldsymbol{p}}_J^T$ becomes more similar and the resulting depth becomes more smooth. During optimization, it is ensured that all the entries of $\boldsymbol{A}$ are positive, so that $\boldsymbol{P}_j\boldsymbol{A}\boldsymbol{h}$ represents a probability. With the aim of solving for (12), stochastic gradient ascent is applied using the partial derivative of (13) with respect to $\boldsymbol{A}$ and $\boldsymbol{\beta}$:

$$\frac{\partial\{\log \widetilde{Pr}(\boldsymbol{y}|\boldsymbol{P}, \boldsymbol{h})\}}{\partial \boldsymbol{A}} = \frac{1}{J}\sum_{j=1}^{J} \frac{\boldsymbol{P}_j^T \widehat{\boldsymbol{p}}_j \boldsymbol{h}^T(\boldsymbol{i}^T\boldsymbol{A}\boldsymbol{h}) - (\widehat{\boldsymbol{p}}_j^T \boldsymbol{P}_j \boldsymbol{A}\boldsymbol{h})\boldsymbol{i}\boldsymbol{h}^T}{[\boldsymbol{i}^T\boldsymbol{A}\boldsymbol{h}]^2} \tag{14}$$

and

$$\frac{\partial\{\log \widetilde{Pr}(\boldsymbol{y}|\boldsymbol{P}, \boldsymbol{h})\}}{\partial \boldsymbol{\beta}} = \frac{1}{|U|}\sum_{(j,k)\in U} \boldsymbol{s}_{j,k}^T \widehat{\boldsymbol{p}}_j \widehat{\boldsymbol{p}}_k^T. \tag{15}$$

$\boldsymbol{A}$ and $\boldsymbol{\beta}$ are randomly initialized, and iteratively updated accordingly. See Section 4.4 for details.

Prediction: Having established $\boldsymbol{A}$ and $\boldsymbol{\beta}$, predicting the posterior probability for new stereo pairs involves solving the Maximum a Posteriori inference on (11). To achieve this, the matrix representations of E_P and E_U are used in (7) and (10) resulting in

$$\widetilde{Pr}(\boldsymbol{d}_j|\boldsymbol{P}_j, \boldsymbol{h}) = \exp\left[\frac{1}{|U|}\boldsymbol{y}^T\boldsymbol{B}\boldsymbol{y} + \frac{1}{N}\boldsymbol{y}^T\boldsymbol{Y}\boldsymbol{A}\boldsymbol{h}\right], \tag{16}$$

The aim is to determine $\boldsymbol{y}$ that maximizes $\widetilde{Pr}(\boldsymbol{y}|x)$ for a pre-computed $\boldsymbol{A}$ and $\boldsymbol{\beta}$ pair.

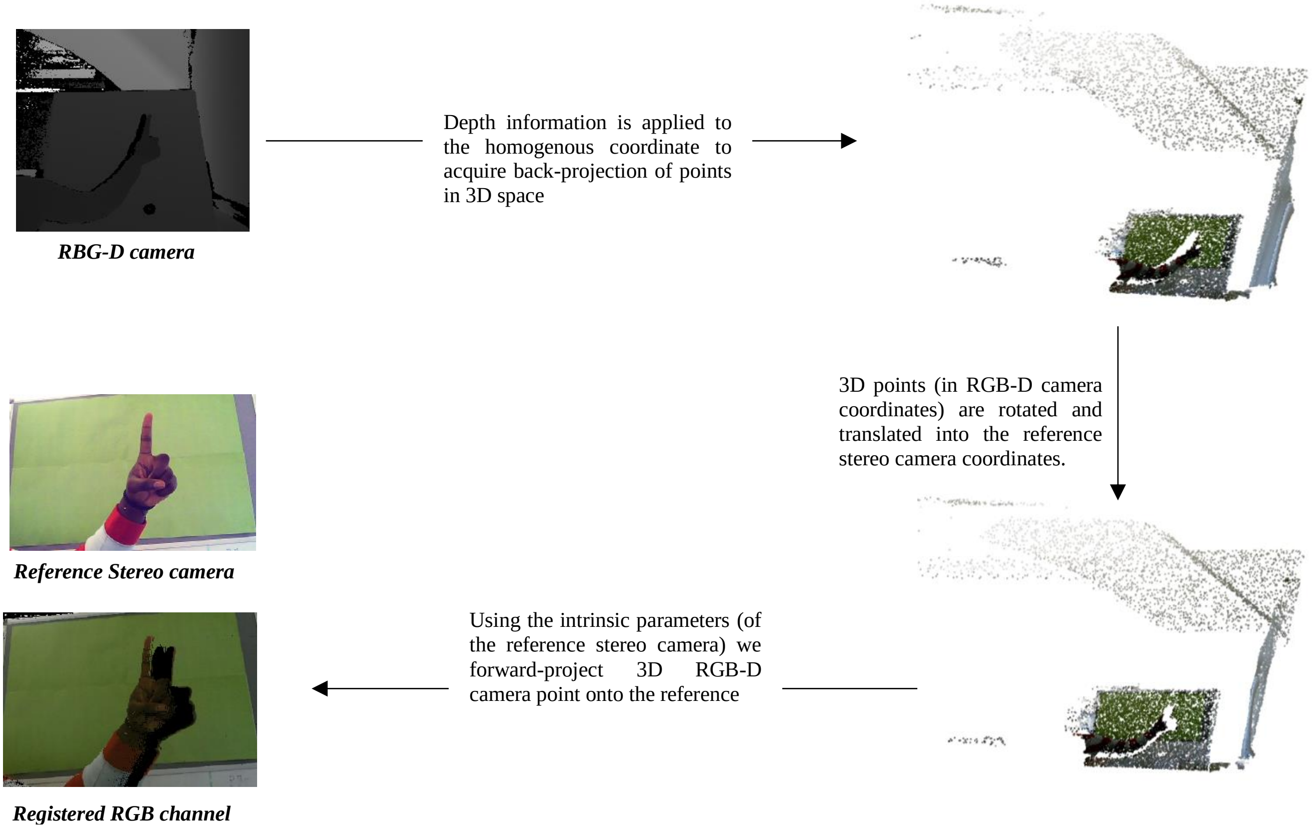
RBG-D camera
Depth information is applied to the homogenous coordinate to acquire back-projection of points in 3D space
3D points (in RGB-D camera coordinates) are rotated and translated into the reference stereo camera coordinates.
Reference Stereo camera
Using the intrinsic parameters (of the reference stereo camera) we forward-project 3D RGB-D camera point onto the reference
Registered RGB channel

Fig. 5: Transferring the depth data from an RGBD camera to establish ground truth depth for the stereo data. Hand poses are captured simultaneously using adjacently positioned calibrated stereo camera and RGBD camera. First, all 2D positions on the RGBD camera are back-projected to 3D. By applying the rotation and translation matrix between the RGBD camera and the reference stereo camera we transform points in the RGBD camera to the camera coordinates of the reference stereo camera. Lastly, we forward project these points onto the reference stereo camera image plane by using its projection matrix. In effect, depth values of the RGBD image are transferred to the reference stereo image, forming ground truth for training the unary term.

$$\boldsymbol{y}^* = \arg\max_{\boldsymbol{y}} \widetilde{Pr}(\boldsymbol{y}|\boldsymbol{P}_j, \boldsymbol{h}) = \arg\max_{\boldsymbol{y}} \frac{1}{|U|}\boldsymbol{y}^T\boldsymbol{B}\boldsymbol{y} + \frac{1}{N}\boldsymbol{y}^T\boldsymbol{Y}\boldsymbol{A}\boldsymbol{h} \tag{17}$$

This is easily derived in closed form by solving for the zeros of the second derivative. Formally,

$$\boldsymbol{y}^* = \frac{|U|}{N}\boldsymbol{B}^{-1}\boldsymbol{Y}\boldsymbol{A}\boldsymbol{h}. \tag{18}$$

$\boldsymbol{y}^*$ represents the concatenated predicted depth probability for all superpixels in an image. The predicted depth level for a superpixel is the depth level with the maximum depth probability.

4 Implementation Details

4.1 Registering Reference Stereo Camera to RGBD Camera

When mapping the matching-cost features to ground truth depth, it was important to establish a database of strong registration between the pairs of data. To achieve this, image and depth acquisition were carried out on both the stereo camera and an RGBD camera, almost adjacently positioned as shown in Fig. 1e. Before capture, the reference stereo camera and the RGB channel of the RGBD camera are stereoscopically calibrated using [16] to establish their respective intrinsic and extrinsic parameters. Images captured from both cameras were undistorted based on the distortion parameters recovered from calibration. First, all points in the RGBD camera plane are back-projected into 3D (by applying its previously calibrated projection matrix and the accompanying depth information). The 3D projection (in the RGBD camera coordinates) is first transformed into the reference stereo camera coordinate (using the calibrated relative rotation and translation information), before being forward-projected into the reference stereo camera plane. Thus, the depth data from the RGBD image is transferred to the reference stereo image (as depicted in Fig. 5). This allows $\{(d_1, \boldsymbol{c}_1)^{(z)}, ..., (d_J, \boldsymbol{c}_J)^{(z)}\}$ to be established for all captured instance of stereo pairs, $\boldsymbol{z}$. In our implementation the Minoru 3D Webcam [*] was used as the stereo camera whilst the Kinect Sensor for Xbox One [†] was used as the RGBD camera.

4.2 Extracted Features

Matching-cost Features, $\boldsymbol{c}_j$: the implementation used five matching cost functions: Sum of Absolute Difference (SAD), Sum of Squared Differences (SSD), Normalized Cross Correlation (NCC), Quantized Census (QC), and Zero-mean Sum of Absolute Differences (ZSAD). The reader is referred to [12] for details on these cost functions. These cost measures were chosen because of their prominence, computation cost, and simplicity. Of course, more complex types and combinations of matching costs could be used. Each of the cost functions was applied under three window sizes: $[7 \times 7]$, $[11 \times 11]$, and $[15 \times 15]$. These window sizes were empirically chosen to demonstrate the low, medium and large range of window sizes.

[*] *http://www.minoru3d.com*

[†] *https://www.xbox.com/en-GB/xbox-one/accessories/kinect*

All combinations of these window sizes and matching costs were used to compare each centroid point in the reference stereo image to 50 potentially matching pixels (selected based on proximity to $\boldsymbol{v}_j$) that lie on the Epipolar line in the corresponding stereo pair. Hence $W = 50$, $G = 3$ and $K = 5$. This resulted in a 750-dimensional matching-cost feature vector being used to regress for the depth at each superpixel.

Holistic Hand Features, $\boldsymbol{h}$: For each captured instance of stereo pairs, three main factors are used to describe the scene. First, the average intensity value of all hand region pixels across all three color channels is considered. This quantifies the skin tone. Second, the aggregative shift of all hand pixels in the reference stereo camera compared to the other stereo camera is computed. This quantifies how far away the hand is from the camera, representing the difference in the average pixel's position for hand region pixels in both cameras. Last, the ratio between the number of hand and non-hand region pixels is computed. This quantifies the size of the hand (if considered relatively to the aggregative shift). This analysis resulted in a six-dimensional holistic hand feature vector (3 color channels values, 2 vector shift values, and 1 ratio of pixels in the hand vs. non-hand regions). Note that the implementation of this technique is not limited to these three factors, the only constraint is that all entries of $\boldsymbol{h}$ must be positive values.

Superpixel Similarity Measure, $\boldsymbol{s}_{j,k}$: To quantify similarities of two neighboring superpixels four measures are used. The first measure is the difference in the average LAB color of both superpixels. The second is the difference in the Local Binary Pattern [31]. The third measure is the difference in the standard deviation of pixels' values in LAB color. Finally, the summed difference in their histograms is examined. In each of these cases, the exponent of the negative Euclidean norm is applied to the resulting difference. E.g. the first entry (LAB difference) $\boldsymbol{s}_{j,k} = e^{-||\boldsymbol{s}_j^{LAB} - \boldsymbol{s}_k^{LAB}||}$, where $\boldsymbol{s}_j^{LAB}$ is the average LAB value for superpixel x_j. This yields a similarity measure vector with a length of four, or $Q = 4$.

4.3 Random Forest

Using the setup described in Section 4.1, 500 instances of hand poses at different distances, from different participants, were captured. Data was captured from 12 participants (6,000 stereo pairs in total) of different skin tone, hand size, and gender. Data from four participants were reserved for testing, and the remaining data

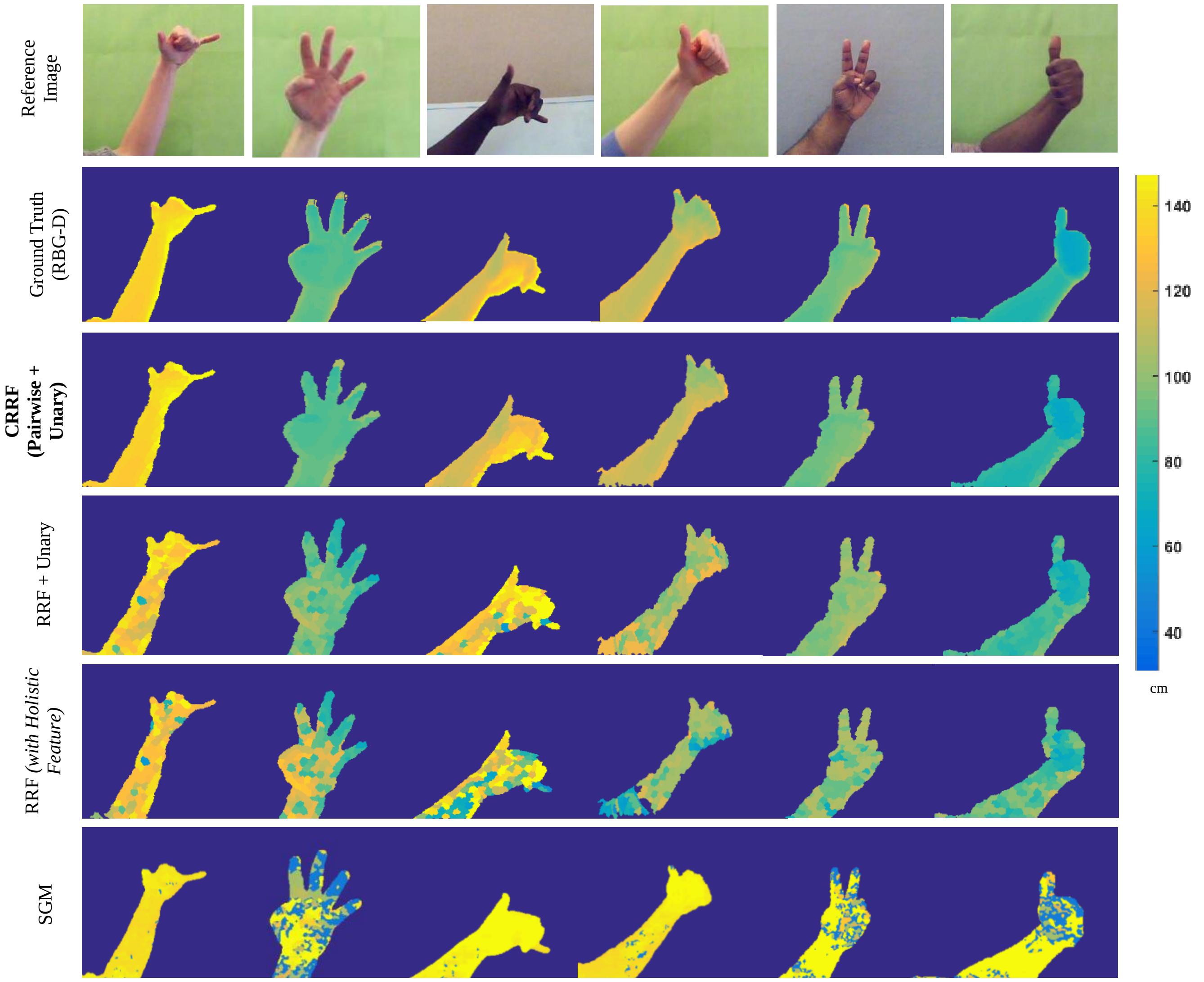

Fig. 6: Qualitative Results using real captured poses. The reference image of the stereo pair is shown in the 1^{st} row and the corresponding groundtruth depth is presented in the 2^{nd} row. The results from the proposed technique are presented in the 3^{rd} row. Results from solely using the unary term with RF are in the 4^{th} row, while recovered depths from RF are presented in the 5^{th} row. The quality of the recovered depth as a result of CRRF is apparent.

(from the other eight participants) was used for training. SLIC segmentation was applied to all reference stereo images, producing approximately 3,000 superpixels per image. Note that only a fraction of these 3,000 superpixels is hand region superpixels. The number of hand superpixels (ranging approximately from 200 to 500 per image capture) depends on the distance between the hand and the camera. In total, roughly 2.5 million superpixels were used in training and evaluating the algorithm. The depth value posterior distribution of the RRF was quantized into 500 bins, i.e. $D = 500$. The depth bin of 500 as it achieves a good balance between the precision of depth prediction and the size of matrix $boldsymbolB$. The depth range of the hand poses in the entire dataset generally ranged from $500mm$ to $1800mm$. Hence, the RRF can predict to a resolution of $(1800mm - 500mm)/500$ bins $= 2.6mm$.

With the focus on the training dataset (from the eight participants), first, all stereo pairs were clustered into six clusters based on the holistic hand feature (using k-means). The training data was divided into two sets (seven participants to one participant). The RRF was trained on the first set (containing data from seven participants) and then the second set (containing data from the remaining participant) was propagated from the trained RRF to acquire the posterior probability matrix, $\boldsymbol{Y}_Z$. This procedure was carried out iteratively for all permutations of seven training and one testing participant(s) in a cross-validation fashion, yielding a set of posterior probabilities $\{\boldsymbol{Y}_1^{(s)}, ..., \boldsymbol{Y}_Z^{(s)}\}$ of stereo images for training participant, s. Note that all $\boldsymbol{Y}_z^{(s)}$ estimations result from testing stereo images of training participant s on an RRF trained on images from all the other seven participants. All $\boldsymbol{Y}_z^{(s)}$ and $\boldsymbol{h}_z^{(s)}$ are subsequently used in the CRRF framework to estimate $\boldsymbol{A}$ and $\boldsymbol{\beta}$. The RRFs were trained in parallel on a cluster with MATLAB using two nodes, each with 20 processors. Each training round (i.e. to train for each posterior $\boldsymbol{Y}_z^{(s)}$) takes approximately 3 - 4 hours. Since eight rounds were needed, training took roughly one day. At test time, based on the MATLAB implementation, the SLIC algorithm runs in 38.23 seconds on average to segment a single reference image. The extraction of the holistic hand and stereo-matching features takes 36.34 seconds on average for each stereo pair. Finally, the propagation of all superpixels and combining the posteriors using $\boldsymbol{\beta}$ execute typically in 185 seconds. Hence testing for the depth a frame of stereo images on the cluster will typically take 260 seconds. Note that the runtime could be considerably reduced in future work by recoding the method in C++ and using GPU techniques.

4.4 Stochastic Gradient Ascent

$\boldsymbol{A}$ and $\boldsymbol{\beta}$ are learned separately by first randomly initializing, with all elements of $\boldsymbol{A}$ being positive. First $\boldsymbol{A}$ is trained for and then $\boldsymbol{\beta}$

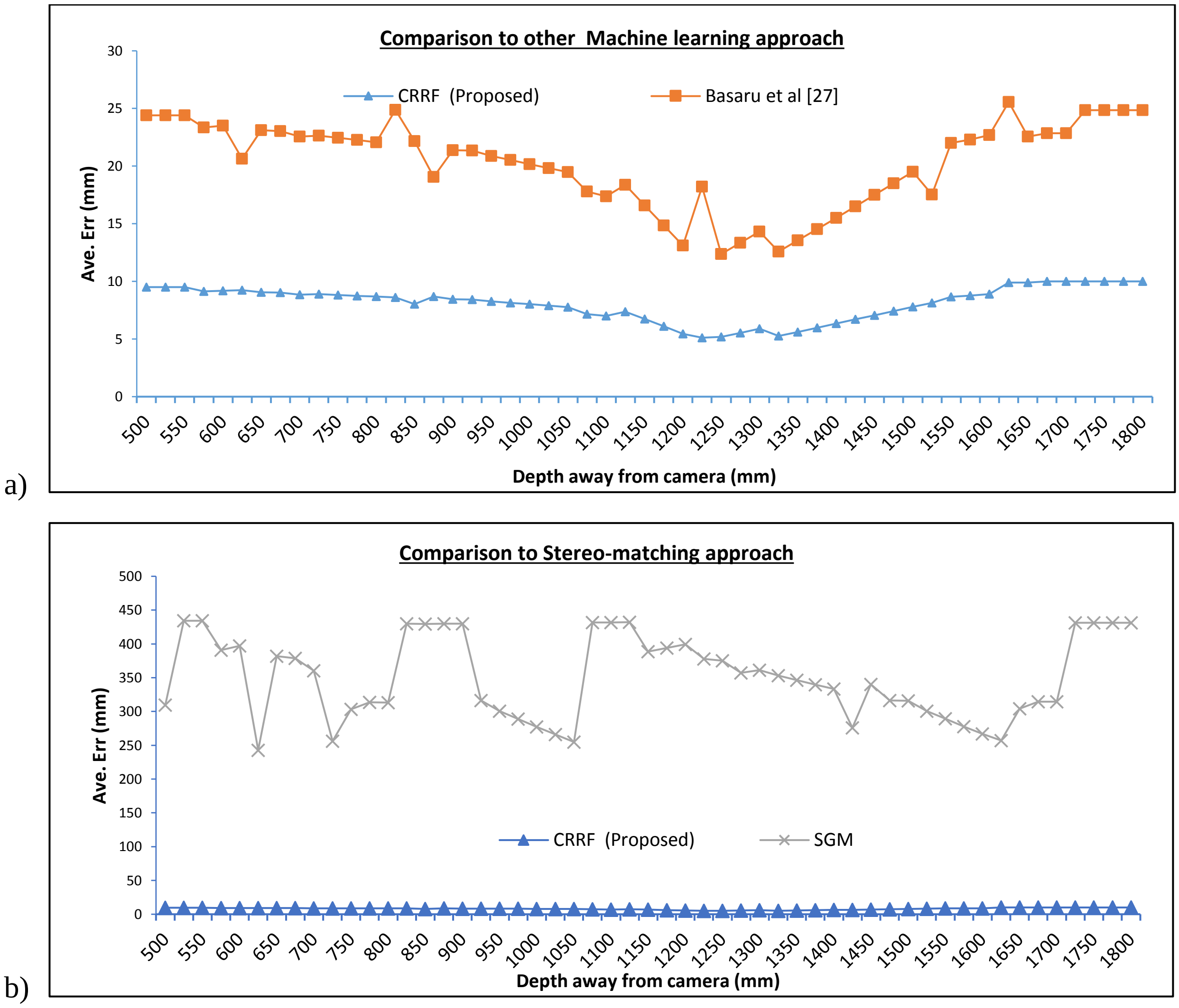

Fig. 7: An illustration of performance over depth levels. The above graphs in Fig. 7a and 7b compare the performance of our method to that of Basaru et al. [27] and to using SGM respectively over different superpixel depth levels.

Table 2 Quantitative comparison of our technique (RF + Pairwise + Unary) against existing work in stereo-matching [24], conventional RRF, and different variants of our technique.

Methods	Superpixel Level Accuracy		Pixel Level Accuracy		Ave. Relative Error	
	t=10mm	t=20mm	t=10mm	t=20mm	per Superpixel	per Pixel
SGM [25]	-	-	0.103	0.132	-	0.772
Basaru et al [24]	-	-	0.455	0.515	-	0.534
RRF	0.599	0.610	0.423	0.492	0.503	0.500
RF (with Holistic Feature)	0.686	0.757	0.610	0.689	0.358	0.353
RF + Unary	0.835	0.885	0.684	0.788	0.229	0.231
CRRF (Pairwise + Unary)	**0.911**	**0.911**	**0.811**	**0.852**	**0.181**	**0.190**

is learnt under a fixed $\boldsymbol{A}$. In both cases, the learning rate was initialized at 12,000. Training was carried out on 100 epochs, reducing the learning rate by 10% every 10 epochs. The decay weight, λ, was set as 0.05.

For greater clarity, we summarize our entire framework in the section, identifying and outlining key features and how they relate in Table 1.

5 Experiments and Results

The approach was validated experimentally, presenting both qualitative (Fig. 6) and quantitative (Table 2) results. Three main comparisons were made, these were prediction solely using RF (with only matching-cost features and with a combination of matching-cost and holistic features); using RF with the unary term framework; as well as a prominent stereo-matching technique (SGM). The results were quantitatively appraised for accuracy by computing the percentage of correctly predicted depth both at superpixel and pixel levels, $\frac{\sum_{p\in N}\mathbb{I}[|d_p^{GT}-d_p|<t]}{N}$, where d_p^{GT} and d_p are the groundtruth and the predicted depth at superpixel (or pixel) p; $\mathbb{I}[]$ is a function that returns 1 for true input and 0 otherwise; and N is the number of hand region pixels/superpixels. The average relative error, $\frac{1}{N}\sum_{p\in N}\frac{|d_p^{GT}-d_p|}{d_p^{GT}}$, was computed to quantitatively evaluate the performance of the test. The following subsections will review the results in Table 2 and Fig. 6 in more detail.

5.1 Stereo-matching Comparison

To validate the machine learning approach, depth recovery (through disparity) from stereo pairs in Dataset B using a prominent stereo-matching technique, SGM was performed. At the time of writing, this was the 9^{th} best performing published stereo-matching technique on the Middlebury stereo evaluation chart [3]. We compare to SGM as it has readily available code and is a performant algorithm offering âĂIJa very good mixture of speed, quality and robustnessâĂİ as described on the authorâĂŹs webpage [2].

We use the same calibration information used in establishing the Epipolar line (in Section 3) to rectify the stereo capture of hand poses (same as those used in the training phase of our evaluation). We then fed the rectified stereo pair into the standard MATLAB implementation of SGM [25] for stereo matching. Stereo baseline and focal length resolved from stereo calibration [16] are combined with the SGM generated disparity to yield the actual distance. We then compute error based on hand pixel regions. The performance is shown in (last row) Fig. 6 and Table 2.

This is an interesting comparison as SGM also applies global optimization. Nonetheless, its poor performance is apparent from Table 2. It provides the least accuracy and the most error in comparison to the rest of the machine learning techniques. The hypothesis here is that this is due to the untextured nature of the hand as well as radiometric differences present in the stereo pair. The SGM technique attempts to universally appraise pixel correspondence by applying a pre-established matching criterion. The untextured nature of the hand and radiometric inconsistencies, in conjunction with the varying skin colors and hand sizes, makes this task hard. This result emphasizes the significance of the proposed approach in that a conventional stereo-matching approach (even one as robust as SGM) performs poorly for skin regions. Further investigation on the performance of the two techniques was performed using pixel level accuracy with a varying threshold. The graphical comparison is presented in Fig. 8. Again, the superiority of CRRF is demonstrated. A significant result is that a high percentage of depth predictions made using the proposed approach are accurate in comparison to SGM. However, as the error threshold gets closer to $8mm$ the percentage drops abruptly. To put this into context, the smallest finger on a hand is typically $10mm$ in width. Hence, at least 81% of the structure of the fingers are mostly discernible. This contrasts with 10.3% in the case of SGM.

5.2 Baseline Comparison

Four baseline comparisons were made. The first was predicting depth solely from the matching-cost feature, using conventional RRF. The results (Table 2) validate the hypothesis that applying a machine learning approach to learning the stereo-matching criteria for determining stereo correspondence is a more effective approach. Using a set of simple stereo-matching criteria and stochastically determining which to use at different tree depths has resulted in almost a 272.7% increase (from 0.132 to 0.492) in pixel level accuracy.

Secondly, the matching-cost feature was augmented by concatenating it with the holistic hand features whilst still regressing with a

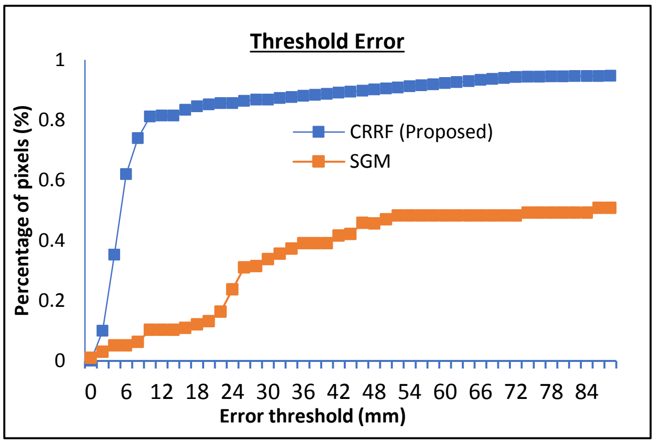

Fig. 8: This graph compare the Pixel Level Accuracy of our technique to SGM as the threshold value varies.

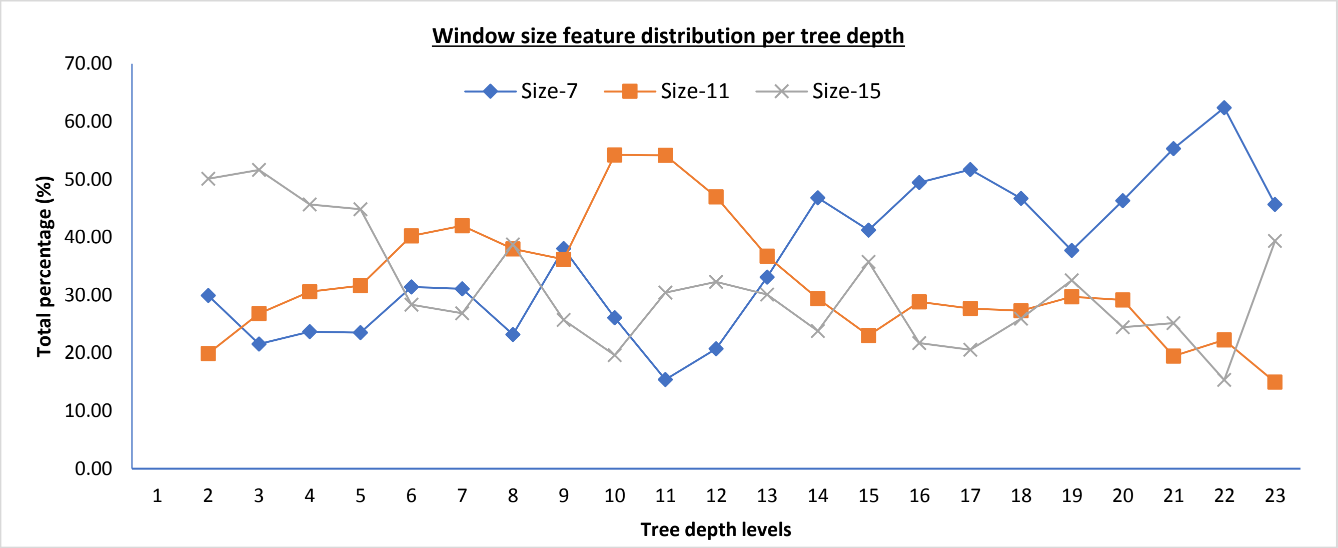

Fig. 9: Graphical illustration of the distribution of feature selection at different depth levels of the regression forest. Specifically, it shows the total percentage of evaluated $\mathbf{7 \times 7}$, $\mathbf{11 \times 11}$, and $\mathbf{15 \times 15}$ window features at different tree depth levels. This corroborates the conjecture that at shallow tree levels the trees are biased to a particular matching criterion. In this case, the larger window sized features ($\mathbf{15 \times 15}$) are evaluated more at shadow tree depth and vice versa.

conventional RRF model. The aim was to specifically investigate the impact of using "expert trees". From Table 2 one can see a notable improvement in the prediction resulting from adding the holistic feature, yielding greater accuracy (0.492 to 0.689) and less relative error (0.500 to 0.353) in both superpixel level and pixel level. However, a much greater increase in accuracy results from using the holistic feature to learn expert trees as opposed to just concatenating it with the stereo-matching feature. This yielded a 50.2% increase in accuracy on average in comparison to the 29.1% increase in accuracy provided by solely concatenating the holistic features.

The last baseline comparison was to investigate the significance of the pairwise term. Recall that the contribution of the pairwise term is to add a smoothing constraint on the depth prediction. This is presented in the qualitative results. The predicted depth is clearly smoother and hence a better representation of the surface of the hand. The quantitative result from Table 2 also conveys the superiority of the prediction made when the pairwise term is applied. Interestingly, the pixel level accuracy is almost as strong as the superpixel level accuracy when the pairwise term is applied. This is again due to the smoothing effect.

5.3 Comparison with [25]

We compared the proposed method with [25], which also applies a regressive random forest to estimate image depth. However, in [25], a single similarity measure (Quantized Census) is used to compute a depth image, and no pairwise term is modeled in the regression that maps a disparity image to a depth image. As the results in Table 2 show, the proposed method, even without the pairwise term, outperforms [25]. We attribute the improved performance of the proposed method to the features used. Unlike [25], which uses a single similarity measure, the proposed method learns the features that best regress the depth using multiple similarity measures, disparity shifts, and window sizes in a concatenated feature vector. Also unlike [25], which uses disparity as an intermediate representation, the proposed method maps directly from the stereo pair to depth. Additionally, our approach to regression is more sophisticated in that we conditionally learn expert trees, which are combined using holistic hand features. Finally, the pairwise term in the proposed model provides additional smoothing constraints that yield superior performance.

5.4 Evaluating Performance vs Depth Range

We further investigated the performance of our technique at different depth levels. To this aim, we compute the average error for pixels of a particular depth range. We present the results in Fig. 7. Fig. 7a, compares the performance of our technique to [25]. It can be observed that in both cases, depth prediction for pixels closer to the camera is relatively poor (higher error). The prediction performance increases for pixels that exist closer to the middle of the depth range. A dip in performance appears again for the further existing pixels. This trend in performance (that is consistent with the machine learning based approach) is not shared by the performance of SGM depth recovery (Fig. 7b). The variation in the performance of SGM is less systematic.

5.5 Evaluating the Coarse-to-fine Conjecture

As stated in the Section 3, the approach was motivated by aiming to implement a coarse-to-fine framework in a machine learning context. This section investigates to what extent the RRF exhibits this coarse-to-fine feature. To do so, during training (of the RRF) all superpixels entering all nodes at each tree depth level were collected and the percentage of superpixels that were evaluated at a particular feature type calculated, keeping in mind that each superpixel that propagates through the RRF possesses a matching-cost feature vector where each of the elements corresponds to a particular window size and matching cost function. Hence, for a superpixel entering a node, the feature position that was evaluated is examined (to determine the split) and tallied. The same applies to matching cost and window size to which the feature corresponds. The results are presented in Table 1a to 1c. Note that the percentile is computed across each depth level. Looking at the tables, it can be noted that the RRF prefers different types of features at different depth levels. For instance, 7×7 and 11×11 window sized SAD features are less evaluated at shallow tree depth (depth levels 4 to 10). The same applies to 15×15 window sized Quantized Census features at deeper tree levels. However, a stronger and more apparent correlation can be observed when the percentiles across window sizes are aggregated (See Fig. 9). An interesting observation pertaining to the correlation between the depth of the trees and the window size is illustrated. At shallow tree depth, the larger sized window (15×15) based feature positions are evaluated more. While in the middle of the latter tree depth smaller window size based feature positions are evaluated more. This is because, at shallow tree depth, where there are higher uncertainty and more variation in the depth of evaluated superpixels, it is advantageous to evaluate affinity based on larger window sizes. In contrast, at deeper tree levels, smaller window sizes are preferred. Some noisy exception to this trend can be observed from the graph, for instance at tree depth level 6 and 7 for the 15×15 window size. This is largely due to the stochastic nature of random forests and to some extent the quantity of data used. We hypothesis that with greater quantity of data, these outliers data point will aggregate to conform with the overall trends.

6 Conclusion

In this paper, we proposed and developed an innovative application of the regression forest technique for resolving depth from stereo images. We present Conditional Regressive Random Forest, a framework that uniquely combines expert trees based on the features of the superpixel whose depth is being predicted. The framework further enforces smoothness constraints as it predicts the depth of superpixels away from the camera. Thus, we have demonstrated the use of a relatively cheap stereo camera rig to generate a high-quality depth image of the hand (see Fig. 1). In achieving this, we established a stereo-depth database of hand captures that is available upon request.

We reiterate that the technique is applicable to other scenarios, including regression problems whereby each data point is not purely independent of other data points. In this case, the Regressive or Classification Random forest can be applied to independently regress for each data point, whilst, the potential dependency between data points can be modeled by the pairwise term. Although the proposed technique is particularly suited to hand estimation because of the holistic features (i.e. classification based on skin tone, hand sizes etc.), nonetheless, it is still applicable to generic scene depth recovery if the expert tree subcomponent is ignored. This would entail solely applying conventional RRF to unary depth estimation (per superpixel) and combining with the pairwise component as discussed above. However, our experimental results demonstrate that the holistic feature is an important component of hand-based stereo depth estimation.

An obvious limitation of the proposed technique is the need of a skin segmentation step that precedes the stereo-matching algorithm. Whilst this does not affect the performance of the technique itself, it will affect the shape of the recovered hand depth. False hand segmentation could be an issue in scenarios where the recovered depth is to be used as a feature for further analysis. For instance, in [24] the feature for pose estimation from depth image is dependent on the shape of the hand. Another potential limitation of this technique is that it quantizes the depth space, limiting the depth sensing reach or resolution. Whilst larger depth sensing reach can be learned by adapting the training set appropriately, this will lead to a computation cost vs. depth reach/resolution trade-off. Since larger depth reach or resolution will require more depth levels (and hence increase in the size of the matrices $\boldsymbol{B}$ and $\boldsymbol{Y}$), the computational expense of the technique increases. A solution to this problem might be to use a logarithmic scale for depth so that less resolution will be given to depth prediction far away (which is often more significant) and vice versa.

RGB cameras have advantages over depth cameras as discussed in the introduction, but computing the depth of a hand using standard stereo algorithms that use a single matching cost function produce inferior results due to ambiguities arising from a lack of texture, and variations in hand size and skin tone. To date, the use of machine learning for hand depth estimation has received little attention, despite the importance of depth estimation for hand gesture and pose estimation in HCI applications. This paper fills this gap by presenting a new state-of-the-art machine learning approach in recovering accurate depth images from stereoscopic images of the hand, and both the qualitative and quantitative results show very promising results.

References

7.1 Websites

[1] 'Microsoft HoloLens Website', https://www.microsoft.com/en-gb/hololens, accessed 18th January 2018

[2] 'DLR - Institute of Robotics and Mechatronics Website', http://www.dlr.de/rm/en/desktopdefault.aspx/tabid-9389/16104_read-39811, accessed 27th January 2018

[3] 'Middlebury Dataset Website', http://vision.middlebury.edu/stereo/data/, accessed 27th May 2017

7.2 Journal articles

[4] Liu K. and Kehtarnavaz N. J. (2016). Real-time Robust Vision-based Hand Gesture Recognition using Stereo Images. In the Journal of Real-Time Image Processing, Volume 11(1)

[5] Fanello S., Keskin C., Izadi S., Kohli P., Kim D., Sweeney D., Criminisi A., Shotton J, Kang S.B., and Paek T., Learning to be a Depth Camera for Close-Range Human Capture and Interaction. In the Journal of ACM (Association for Computing Machinery) Transactions on Graphics, Volume 33 (4)

[6] Phung, S., Bouzerdoum, A., and Chai, D., (2005). Skin Segmentation Using Color Pixel Classification: Analysis and Comparison. In the IEEE Transactions on Pattern Analysis and Machine Intelligence, Volume 27 (1)

[7] Hasan, M.M.,and Mishra, P.K. (2012). Superior Skin Color Model using Multiple of Gaussian Mixture Model. In the British Journal of Science, Volume 6 (1)

[8] Hasan, M.M., and Mishra, P.K. (2013). Novel Algorithm for Skin Color Based Segmentation Using Mixture of GMMs. In the Signal and Image Processing International Journal. Volume 4 (4)

[9] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Susstrunk, (2012). SLIC Superpixels Compared to State-of-the-Art Superpixel Methods, In the IEEE Transactions on Pattern Analysis and Machine Intelligence, Volume 34 (11)

[10] Ye M., Shen Y., and Du C. (2016). Real-Time Simultaneous Pose and Shape Estimation for Articulated Objects Using a Single Depth Camera, In the IEEE Transactions on Pattern Analysis and Machine Intelligence, Volume 38 (8)

[11] Ding M., and Fan G. (2016). Articulated and Generalized Gaussian Kernel Correlation for Human Pose Estimation, In the IEEE Transactions on Image Processing, Volume 25 (2)

7.3 Conference Paper

[12] Basaru, R., Alonso, E., Child, C., and Slabaugh, G., (2014). Quantized Census for Stereoscopic Image Matching. In Proc. of the 3DV Conference: Workshop, Dynamic Shape Measurement and Analysis. Dec 2014, Tokyo, Japan

[13] HirschmÃijller H. & Scharstein D. (2007). Evaluation of Cost Functions for Stereo Matching. In Proc. of the IEEE Conference on Computer Vision and Pattern Recognition. June 2007, Minneapolis, Minnesota, USA

[14] Shotton, J., Fitzgibbon, A., Cook, M., Sharp, T., Finocchio, M., Moore, R., Kipman, A., and Blake, A. (2011). Real-Time Human Pose Recognition in Parts from Single Depth Images. In Proc. of the IEEE Conference on Computer Vision and Pattern Recognition. June 2011, Colorado Springs, Colorado, USA

[15] Liu F., Gould S. and Shen C. (2014). Deep Convolutional Neural Fields for Depth Estimation from a Single Image. In Proc. of the

IEEE Conference on Computer Vision and Pattern Recognition. June 2014, Columbus, Ohio, USA

[16] Zhang, Z. (1999). Flexible Camera Calibration by Viewing a Plane from Unknown Orientations. In Proc. of the International Conference on Computer Vision. September 1999, Corfu, Greece

[17] Thacker N., Aherne F. and Rockett P. (1997). The Bhattacharyya Metric as an Absolute Similarity Measure for Frequency Coded Data. Techniques in Pattern Recognition. June 1997, Prague, Czech Republic

[18] Liu B., Gould S. and Koller D. (2010). Single Image Depth Estimation from Predicted Semantic Labels. In Proc. IEEE Conference for Computer Vision and Pattern Recognition. June 2010, San Francisco, California, USA

[19] Saxena A., Chung S.H. and Ng A. Y. (2005). Learning Depth from Single Monocular Images. In Proc. of the Advances in Neural Information Processing Systems. Vancouver, British Columbia, Canada

[20] Eigen D., Puhrsch C., and Fergus R. (2014). Depth Map Prediction from a Single Image Using a Multi-Scale Deep Network. In Proc. of the Advances in Neural Information Processing Systems. Montreal, Quebec, Canada

[21] Oikonomidis I., Kyriazis N., and Argyros A.A. (2011). Full Dof Tracking of a Hand Interacting with an Object by Modelling Occlusions and Physical constraints. In Proc. of the IEEE Conference on Computer Vision and Pattern Recognition. June 2011, Colorado Springs, Colorado, USA

[22] Oikonomidis I., Kyriazis N., and Argyros A.A. (2012) Tracking the Articulated Motion of Two Strongly Interacting Hands. In Proc. of the IEEE Conference on Computer Vision and Pattern Recognition. June 2012, Providence, Rhode Island, USA

[23] Grzeszczuk R., Bradski G., Chu M.H., and Bouguet J.Y. (2000). Stereo Based Gesture Recognition Invariant to 3d Pose and Lighting. In Proc. of the IEEE Conference on Computer Vision and Pattern Recognition. June 2000, Hilton Heads Island, South Carolina, USA

[24] Basaru, R., Alonso, E., Child, C., and Slabaugh, G., (2016). HandyDepth: Example-based Stereoscopic Hand Depth Estimation using Eigen Leaf Node Features. In Proc. of the IWSSIP International Conference. May 2016, Bratislava, Slovakia

[25] Hirschmuller, H. (2005). Accurate and Efficient Stereo Processing by Semi-Global Matching and Mutual Information. In Proc. of the IEEE Conference on Computer Vision and Pattern Recognition. June 2005, San Diego, California, USA

[26] Basaru, R., Child, C., Alonso, E., and Slabaugh, G., (2017). Conditional Regressive Random Forest Stereo-based Hand Depth Recovery. In Proc. of International Conference on Computer Vision: HANDS Workshop, Oct 2017, Venice, Italy

[27] Payet, N., and Todorovic, S., (2010) Random Forest Random Field. In Proc. of the Advances in Neural Information Processing Systems. Vancouver, British Columbia, Canada

[28] Sun, M., Kohli, P., and Shotton, J. (2012). Conditional Regression Forests for Human Pose Estimation. In Proc. of the IEEE Conference on Computer Vision and Pattern Recognition. June 2012, Providence, Rhode Island, USA

7.4 Book, book chapter and manual

[29] Criminisi, A., and Shotton, J. (2013). Decision Forests for Computer Vision and Medical Image Analysis. Berlin: Springer.

[30] Murphy K.P. (2012) Machine Learning - A Probabilistic Perspective. Cambridge: MIT Press

[31] Pietikainen, M., Hadid, A., Zhao, G., and Ahonen, T. (2011) Computer Vision Using Local Binary Patterns. London: Springer-Verlag

8 Appendices

Table 3 Percentile distribution of evaluated features based on their window size and matching cost function at different depth levels.

7 × 7 window

		Cost Function				
		SAD	SSD	NCC	ZSAD	QC
RF Depth Level	1	15.51	1.56	0.36	2.77	1.17
	2	14.49	2.20	0.08	3.34	9.85
	3	13.51	5.74	2.27	0.00	0.02
	4	0.28	11.57	1.22	7.90	2.70
	5	0.08	12.82	1.96	6.51	2.15
	6	0.02	17.91	2.69	8.76	2.01
	7	2.78	14.36	2.95	9.50	1.50
	8	1.93	4.97	3.60	11.49	1.21
	9	13.60	1.34	0.31	21.80	1.00
	10	13.94	2.11	0.08	3.19	9.41
	11	9.66	4.10	1.62	0.00	0.02
	12	0.25	10.12	1.07	6.91	2.36
	13	0.05	18.10	2.77	9.19	3.03
	14	0.03	26.73	4.02	13.08	3.00
	15	3.68	19.03	3.91	12.59	1.99
	16	12.53	13.12	14.37	2.56	6.89
	17	21.54	11.71	7.17	2.86	8.43
	18	12.22	15.11	7.28	0.00	12.11
	19	10.55	13.79	0.00	0.29	13.10
	20	10.97	17.18	0.92	0.31	16.98
	21	11.85	21.17	0.00	0.00	22.31
	22	10.71	0.00	10.52	39.48	1.70
	23	9.47	0.00	30.67	0.00	5.51

(a)

11 × 11 window

		Cost Function				
		SAD	SSD	NCC	ZSAD	QC
RF Depth Level	1	1.55	13.53	4.75	2.77	0.27
	2	4.38	1.35	1.01	3.34	9.85
	3	0.14	0.13	7.68	5.48	13.38
	4	0.67	15.96	6.59	5.83	1.57
	5	0.11	10.21	7.65	6.78	6.88
	6	0.00	12.96	10.05	8.90	8.37
	7	0.00	13.45	10.70	9.48	8.40
	8	0.00	15.72	12.76	0.00	9.54
	9	0.12	12.86	8.70	7.71	6.82
	10	1.98	8.09	9.90	8.77	30.89
	11	4.94	2.97	9.04	8.01	29.20
	12	0.27	7.73	18.96	0.86	19.15
	13	17.77	2.70	0.10	4.10	12.08
	14	18.43	7.82	3.10	0.00	0.03
	15	0.37	15.13	1.51	2.51	3.53
	16	0.00	15.76	2.42	8.00	2.64
	17	0.02	15.81	2.38	7.74	1.77
	18	0.77	12.41	3.24	10.91	0.00
	19	8.40	5.62	3.25	10.99	1.46
	20	0.78	9.86	4.02	13.61	0.90
	21	13.19	0.00	3.15	0.00	3.15
	22	10.33	0.00	10.22	0.00	1.70
	23	9.45	0.00	0.00	0.00	5.51

(b)

15 × 15 window

		Cost Function				
		SAD	SSD	NCC	ZSAD	QC
RF Depth Level	1	15.91	14.15	15.51	2.77	7.43
	2	14.84	13.69	8.38	3.34	9.85
	3	13.51	16.70	8.05	0.00	13.38
	4	11.59	15.82	3.59	0.31	14.38
	5	10.61	16.62	0.89	0.30	16.43
	6	13.37	11.34	0.30	2.33	0.98
	7	12.99	1.98	0.07	3.00	8.83
	8	12.05	5.12	2.03	19.57	0.02
	9	0.31	12.58	1.33	8.59	2.93
	10	0.04	0.00	1.96	6.51	11.14
	11	0.01	14.01	2.10	6.85	7.46
	12	2.47	12.73	2.62	13.17	1.33
	13	2.51	6.45	4.67	14.90	1.57
	14	19.90	1.96	0.45	0.00	1.47
	15	17.34	2.62	0.10	3.97	11.71
	16	13.62	5.78	2.29	0.00	0.02
	17	7.42	2.04	1.15	7.43	2.54
	18	21.40	4.31	0.23	0.00	0.00
	19	0.00	0.00	11.99	20.56	0.00
	20	2.18	0.00	5.57	16.72	0.00
	21	4.18	0.00	8.40	12.60	0.00
	22	10.22	0.00	0.00	5.11	0.00
	23	9.45	0.00	29.93	0.00	0.00

(c)