

PROPEL: Probabilistic Parametric Regression Loss for Convolutional Neural Networks

Muhammad Asad
Imagination Technologies
Kings Langley, UK
masadc@gmail.com

Rilwan Basaru
Onaria Technologies
London, UK
remi@onariatech.com

S M Masudur Rahman Al Arif
ASML
Veldhoven, Netherlands
masudur.al.arif@asml.com

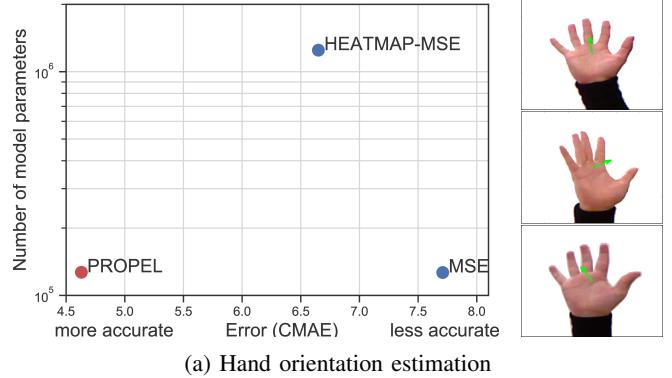
Greg Slabaugh
City, University of London
London, UK
greg.slabaugh@gmail.com

Abstract—In recent years, Convolutional Neural Networks (CNNs) have enabled significant advancements to the state-of-the-art in computer vision. For classification tasks, CNNs have widely employed probabilistic output and have shown the significance of providing additional confidence for predictions. However, such probabilistic methodologies are not widely applicable for addressing regression problems using CNNs, as regression involves learning unconstrained continuous and, in many cases, multi-variate target variables. We propose a PRObabilistic Parametric rEgression Loss (PROPEL) that facilitates CNNs to learn parameters of probability distributions for addressing probabilistic regression problems. PROPEL is fully differentiable and, hence, can be easily incorporated for end-to-end training of existing CNN regression architectures using existing optimization algorithms. The proposed method is flexible as it enables learning complex unconstrained probabilities while being generalizable to higher dimensional multi-variate regression problems. We utilize a PROPEL-based CNN to address the problem of learning hand and head orientation from uncalibrated color images. Our experimental validation and comparison with existing CNN regression loss functions show that PROPEL improves the accuracy of a CNN by enabling probabilistic regression, while significantly reducing required model parameters by $10\times$, resulting in improved generalization as compared to the existing state-of-the-art.

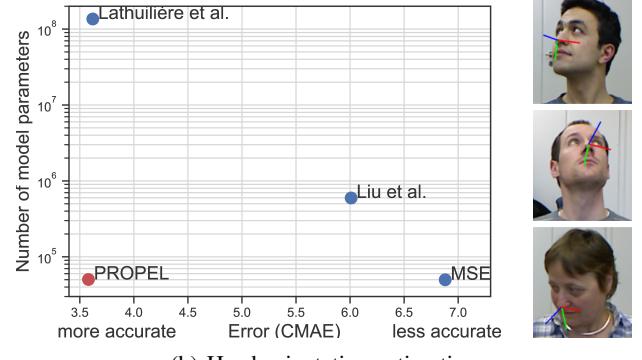
I. INTRODUCTION

Convolutional Neural Networks (CNNs) are enabling major advancements in a range of machine learning problems. For classification tasks, the existing state-of-the-art benefits from probability distributions for target class prediction [1, 2, 3, 4]. These distributions provide additional confidence information that can be useful to determine the level of uncertainty in a given prediction and help resolve ambiguous model predictions [5]. Furthermore, for complex learning problems where dividing a given learning task into smaller subtasks can help, probabilistic models provide a flexible framework for combining the predictions from multiple models [5, 6].

Despite widespread application of CNNs for probabilistic classification, little emphasis has been made for unconstrained probabilistic regression. Regression using a CNN has been restricted to target space learning using Mean Squared Error (MSE) that does not provide a probabilistic output [10, 11]. In some existing methods, non-parametric probability heatmaps representing target variables are directly learned using MSE [12, 13, 14]. In order to generate target heatmap distributions,



(a) Hand orientation estimation



(b) Head orientation estimation

Fig. 1: Number of model parameters (y-axis, note the log scale) vs accuracy (x-axis) for (a) hand [5] and (b) head orientation regression [7]. PROPEL achieves state-of-the-art accuracy for both tasks, with significantly fewer required model parameters as compared to methods Lathuilière et al. [8], Liu et al. [9], Mean Squared Error (MSE) and HEATMAP-MSE.

these methods make two assumptions. First, that the target continuous variables can be discretized into heatmap bins without significant loss in accuracy. Second, these variables can be constrained within a specific domain covered by heatmaps. Although necessary, these assumptions limit application of non-parametric probabilistic regression within a specific domain. Moreover, such methods require the output layer of a CNN to represent significantly higher dimensional heatmaps. This contributes to the complexity of the model,

This work was undertaken when authors were at City, University of London

requiring larger number of required parameters, increased learning time and training data.

We propose a novel PRObabilistic Parametric rEgression Loss (PROPEL) that enables a CNN model to learn parametric regression probabilities. Using PROPEL a CNN model can learn parameters of a Mixture of Gaussians distributions in the target space. PROPEL is fully differentiable with an analytic closed-form solution to integrals, allowing it to be used for end-to-end training of existing CNN regression architectures. Moreover, the proposed loss is generalizable for addressing different levels of complexity within prediction probabilities as well as the number of dimensions in multi-variate regression problems. As the output is parameterized by a Mixture of Gaussians, the proposed layer reduces the number of learned parameters as compared to previously used methods based on directly learning probability heatmaps. In this work, we demonstrate PROPEL’s ability to enable probabilistic regression by addressing the problem of color image-based hand and head orientation regression [5, 7]. Both tasks involve learning of ambiguous cases due to symmetries, where multiple orientations can have similar feature representation, which benefit from utilizing probabilistic regression [5]. Our experimental results show that PROPEL-based CNN achieve state-of-the-art accuracy, while reducing required model parameters by $10\times$. These results are summarized in Fig. 1, showing the accuracy and model parameters trade-off for PROPEL and existing state-of-the-art methods.

The main contributions of this paper are:

- We introduce PROPEL which, to the best of our knowledge, is the first fully differentiable loss layer for enabling unconstrained probabilistic parametric regression using CNNs. A novel derivation is provided for the proposed loss using a Mixture of Gaussians distributions;
- We present a framework that is generalizable for different number of target dimensions and has the ability to model complex multimodal probabilities with additional Gaussians in the Mixture of Gaussians;
- We provide experimental validation and comparisons with existing methodologies and report that the proposed loss outperforms existing state-of-the-art, while reducing the number of required model parameters by $10\times$.

II. RELATED WORK

Probabilistic Regression using CNN. Previous work has mostly been focused on exploring non-parametric probabilistic regression for CNNs [15, 12, 13, 16, 14]. These methods first generate the ground truth probability heatmap distributions in the target space. A CNN is then trained using a Mean Squared Error (MSE) loss to learn the mapping of input images onto the heatmaps. In [14] Tompson et al. generated and employed single-view 2D heatmaps for hand joints localization using depth images as input. Ge et al. [13] extended [14] to use multi-view CNN, learning 2D heatmaps for three projections of depth images. Similar methods were also proposed for human pose estimation using a CNN [12, 16]. Recent work has also explored the use of 3D heatmaps for hand joint

localization [17, 18]. Moon et al. [17] proposed to use 3D convolutions to learn mapping of 3D voxelized depth image onto hand joint locations represented as 3D heatmaps. In contrast to using MSE, [15] learns heatmap distributions by utilizing the Bhattacharyya coefficient (BC) as the loss function. Probabilistic interpretation of Euclidean loss has also been previously explored in [19].

Lathuilière et al. [8] jointly learned both a CNN model, for representation learning, and a Gaussian mixture of linear inverse regressions by utilizing a modified Expectation Maximization (EM) algorithm. Similar to PROPEL, this approach enabled probabilistic regression using CNNs. Crucially, this method required a carefully designed initialization, that included a pre-trained CNN and clustering of data.

Due to computational complexity, non-parametric heatmap distributions only work in problems where the target space can be fully defined within a finite domain, such as the 2D/3D domain used in hand or body joint localization [12, 13, 16, 14, 17, 18]. Furthermore, the use of heatmaps require these methods to assume that the continuous target variable can be discretized into heatmap bins without significant accuracy loss. While such assumptions prove useful for introducing and benefiting from the probabilistic regression in CNNs, they limit generalization of these methods and their application to other regression problems. Moreover, the number of model parameters required to directly learn a non-parametric probability heatmap distribution are much higher. The use of heatmaps also limits target variables to lower dimensional problems, e.g. addressing 3D regression problem using 3D heatmaps [17, 18] or by decomposing 3D targets into multiple 2D heatmaps [12, 13, 14]. Additionally, for the joint localization problem, the methods assume that each target joint vary independently, which could have adverse impact on addressing the underlying regression problem [20].

PROPEL addresses the limitations of existing non-parametric probabilistic regression methods by utilizing a parametric loss function that does not require the target space to be constrained. Furthermore, PROPEL is generalizable in terms of target variable dimension and prediction probability complexity. As indicated by our experimental validation and summarized in Fig. 1, PROPEL generalizes well when trained on a smaller dataset, while also helping to significantly reduce the number of learned model parameters.

Hand orientation Estimation. Image-based hand orientation regression has only been applied in [21, 20, 5]. [21] utilized two single-variate Random Forest (RF) regressors based on an assumption that the orientation angles vary independently. Later, [20] used a multi-layered Random Forest regression method that utilized multi-variate regressors to regress the orientation angles together to exploit target variable dependence. Similarly, [5] presented a staged probabilistic regressor, which learned multiple expert Random Forests in stages. Yang et al. [22] proposed a method for localizing hand and inferring only in-plane rotation in images containing multiple hand.

All existing image-based hand orientation regression methods utilize hand-crafted features. In this paper, we show

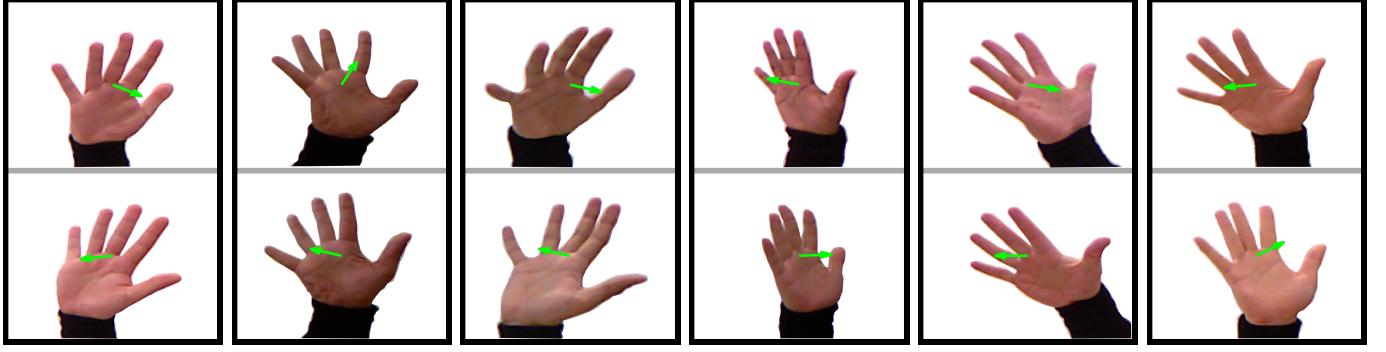


Fig. 2: Symmetry problem due to depth ambiguity in hand orientation. This dataset shows image pairs with different orientations (shown with \uparrow) but similar hand shapes. This motivates the need for multimodal probability distributions in PROPEL.

that the task of hand orientation regression can benefit from the representation learning capability of a CNN. Utilizing MSE, we show that a CNN model, without any probabilistic output, can significantly outperform existing methods that use hand-crafted features. We use this model as baseline for our experimental validation and demonstrate that the proposed probabilistic parametric regression loss can outperform existing regression loss functions for CNNs. This is particularly due to PROPEL’s ability to handle ambiguity in hand orientation datasets by utilizing probabilistic distributions to comprehensively represent the predictions.

Head orientation Estimation. A number of existing methods address head orientation regression using color images [23, 24, 9, 8]. Drouard et al. [23] proposed Gaussian mixture of locally-linear mapping model. This method learned the mapping of HOG-based features onto head orientation, additionally providing probabilistic output. Later, Lathuilière et al. [8] improved [23] by utilizing a pre-trained CNN for feature extraction. This method, however, relied on a modified EM algorithm for fine-tuning a CNN and learning the mapping of features onto target head orientations. CNN-based head orientation regressors were proposed in [24, 9]. Both methods learned using MSE loss, where [9] utilized an additional synthetic dataset for improving the model. In our experimental validation, we compare against these existing methods and show that, due to the probabilistic reasoning, PROPEL can achieve state-of-the-art accuracy for head orientation estimation, while requiring significantly less learned model parameters.

III. METHOD

We now describe the details of our proposed PRObabilistic Parametric rEgression Loss (PROPEL). First, we introduce the loss function and the corresponding parametric probabilities. This is followed by the derivation of a novel analytic closed-form solution to the integrals within our loss function that are required for computing the loss over an unconstrained real space. Lastly, we provide details of the partial derivatives of our loss function which allows PROPEL to be integrated with existing CNN architectures using backpropagation.

A. Probabilistic Parametric Regression Loss

We expand on [25] by introducing a regression loss function for measuring loss between ground truth (GT) and predicted parametric probability distributions. The existing measures, such as Bhattacharyya coefficient (BC) and Kullback-Leibler (KL) divergence, are tractable when distributions are *unimodal*, i.e. each consisting of a single Gaussian. It is well-known that these measures between *multimodal* Mixture of Gaussians have no analytic solution [26] and at best, one must resort to approximation. However, to address modelling of complex multimodal probabilistic distributions, and handling ambiguous predictions (see Fig .6), Mixture of Gaussians are essential. This motivates the use of the measure from [25] for proposing PROPEL, as this measure has an analytic closed-form solution when distributions are Mixture of Gaussians. We show, for the first time in this paper, how to analytically address neural network regression using a Mixture of Gaussians loss. PROPEL is fully differentiable, enabling us to determine both the loss and gradient of a predicted model distribution P_m with respect to a ground truth (GT) distribution P_{gt} . Let $\mathbf{x} = \{x_1, x_2, \dots, x_n\}^\top \in \mathbb{R}^n$ define the target prediction space with n dimensions, the proposed loss can be defined for \mathbf{x} as:

$$L = -\log \left[\frac{2 \int P_{gt} P_m d\mathbf{x}}{\int (P_{gt})^2 + P_m^2 d\mathbf{x}} \right], \quad (1)$$

where P_{gt} is the GT n -dimensional Gaussian probability density function (PDF) defined as:

$$P_{gt}^k = \frac{e^{-\frac{1}{2} \left[\frac{(x_1 - \mu_{x_{1k}})^2}{\sigma_{x_{1k}}} + \dots + \frac{(x_n - \mu_{x_{nk}})^2}{\sigma_{x_{nk}}} \right]}}}{(\sqrt{2\pi})^n \sqrt{\sigma_{x_{1k}} \dots \sigma_{x_{nk}}}}, \quad (2)$$

where k represents a sample selected from a given dataset, $\mu_{x_{1k}}, \mu_{x_{2k}}, \dots, \mu_{x_{nk}}$ are the GT labels and $\sigma_{\mu_{x_{1k}}}, \sigma_{\mu_{x_{2k}}}, \dots, \sigma_{\mu_{x_{nk}}}$ are the GT variances associated with P_{gt} .

In addition to the GT distribution P_{gt} , PROPEL requires a model PDF P_m , parameters of which are learned by a CNN using PROPEL loss from Equation 1. The choice of P_m determines the ability of a trained model to handle complex variations in target space for a given dataset. Our work is

motivated by the limitation of existing regression techniques for addressing ambiguity within a given dataset. Consider the problem of learning hand orientations from uncalibrated color images. As noted in existing literature [20] and shown in Fig. 2, the absence of depth information results in ambiguity, where multiple symmetrically opposite hand orientations have similar color images. To address such cases, a unimodal probabilistic distribution proves insufficient (the same is true for probabilistic interpretation of MSE). In contrast, utilizing a multimodal distribution enables the regressor to address such ambiguities by learning to infer multiple hypotheses [5, 20]. In this work, we choose Mixture of Gaussians as the model PDF P_m as it facilitates in keeping our derivations simple, while also enabling us to model complex multimodal distributions necessary to address the ambiguity within a dataset. Moreover, as compared to existing state-of-the-art which uses non-parametric heatmaps, the Mixture of Gaussians only require a fraction of parameters (means and variances) to represent the model distribution. This reduces the number of model parameters as well as the overall complexity of a CNN model (see Fig. 1). Additionally, the number of Gaussians in the Mixture of Gaussians provide flexibility to learn complex probability distributions, resulting in better accuracy while keeping the derivation consistent. For a target space with n dimensions, P_m is defined as:

$$P_m = \frac{1}{I} \sum_{i=1}^I P_i = \frac{1}{I} \sum_{i=1}^I \frac{e^{-\frac{1}{2} \left[\frac{(x_1 - \mu_{x_{1i}})^2}{\sigma_{x_{1i}}} + \dots + \frac{(x_n - \mu_{x_{ni}})^2}{\sigma_{x_{ni}}} \right]}}{(\sqrt{2\pi})^n \sqrt{\sigma_{x_{1i}} \dots \sigma_{x_{ni}}}}, \quad (3)$$

where P_i represents an individual i^{th} Gaussian within a Mixture of Gaussians, $\mu_{x_{1i}} \dots \mu_{x_{ni}}$ and $\sigma_{x_{1i}} \dots \sigma_{x_{ni}}$ are model parameters inferred as the output of a CNN network and I is the total number of Gaussians in P_m .

Our model distribution does not include covariances since in our experimental validation we found variances to be sufficient for outperforming the state-of-the-art methods. Comparing model distribution P_m to a Gaussian Mixture Model (GMM), the Gaussians P_i are equally weighted by $\frac{1}{I}$, whereas in a GMM, each Gaussian is given its own weight. However, we note that in our formulation, the Gaussians have different shapes and *heights* resulting from their differing standard deviations. Since the denominator in Equation 3 includes standard deviation terms of the form $\sqrt{\sigma_{x_{1i}} \dots \sigma_{x_{ni}}}$, modes that have less variance (i.e., have more confidence) will have higher peaks in the output. This can be seen in Fig. 6 (b), where two modes are “competing” for the solution, but the one with less variance has a higher peak and is selected as the solution. By eliminating the weights of a GMM, PROPEL can produce analytic derivatives for back-propagation in deep networks, and at the same time result in giving higher weightage to the peaks in more confident areas of the solution space.

The next section provides a novel analytic closed-form solution of the loss function L such that we can evaluate the integrals over the continuous domain $-\infty$ to $+\infty$.

B. Analytic Solution to Integrals

Given the loss function L , we can substitute the probability density functions (PDFs) for model P_m and GT distribution P_{gt} in Equation 1 and simplify.

$$L = -\log \left[\frac{2 \int P_{gt} P_m d\underline{x}}{\int (P_{gt})^2 + (P_m)^2 d\underline{x}} \right], \quad (4)$$

$$= -\log \left[2 \int P_{gt} P_m d\underline{x} \right] + \log \left[\int P_{gt}^2 d\underline{x} + \int P_m^2 d\underline{x} \right], \quad (5)$$

$$\begin{aligned} &= -\log \underbrace{\left[\frac{2}{I} \sum_{i=1}^I G(P_{gt}, P_i) \right]}_{T1} \\ &\quad + \log \underbrace{\left[H(P_{gt}) + \frac{1}{I^2} \sum_{i=1}^I H(P_i) + \frac{2}{I^2} \sum_{i < j} G(P_i, P_j) \right]}_{T2}, \end{aligned} \quad (6)$$

where the functions $G(P_i, P_j)$ and $H(P_i)$ represent the analytic solutions to the integrals $\int P_i P_j d\underline{x}$ and $\int P_i^2 d\underline{x}$, respectively. P_i and P_j are two multi-variate Gaussian distributions. Both $G(,)$ and $H()$ are defined as follows¹:

$$G(P_i, P_j) = \int P_i P_j d\underline{x}, \quad (7)$$

$$= \frac{e^{\left[\frac{2\mu_{x_{1i}}\mu_{x_{1j}} - \mu_{x_{1i}}^2 - \mu_{x_{1j}}^2}{2(\sigma_{x_{1i}} + \sigma_{x_{1j}})} + \dots + \frac{2\mu_{x_{ni}}\mu_{x_{nj}} - \mu_{x_{ni}}^2 - \mu_{x_{nj}}^2}{2(\sigma_{x_{ni}} + \sigma_{x_{nj}})} \right]}}{(\sqrt{2\pi})^n \sqrt{(\sigma_{x_{1i}} + \sigma_{x_{1j}}) \dots (\sigma_{x_{ni}} + \sigma_{x_{nj}})}}. \quad (8)$$

$$H(P_i) = \int P_i^2 d\underline{x} = \frac{1}{(2\sqrt{\pi})^n \sqrt{\sigma_{x_{1i}} \dots \sigma_{x_{ni}}}}. \quad (9)$$

Next, we show how the loss L from Equation 6 can be used alongside existing CNN architectures that use backpropagation for training.

C. Optimization

In this section we present the partial derivatives of L with respect to model parameters $\mu_{x_{ni}}, \sigma_{x_{ni}}$. These can be used for end-to-end training of a CNN using backpropagation. The partial derivatives of L are:

$$\frac{\partial L}{\partial \mu_{x_{ni}}} = -\frac{1}{T1} \left[\frac{\partial G(P_{gt}, P_i)}{\partial \mu_{x_{ni}}} \right] + \frac{1}{T2} \left[\frac{2}{I^2} \sum_{i < j} \frac{\partial G(P_i, P_j)}{\partial \mu_{x_{ni}}} \right], \quad (10)$$

$$\frac{\partial L}{\partial \sigma_{x_{ni}}} = -\frac{1}{T1} \left[\frac{\partial G(P_{gt}, P_i)}{\partial \sigma_{x_{ni}}} \right] + \frac{1}{T2} \left[\frac{1}{I^2} \frac{\partial H(P_i)}{\partial \sigma_{x_{ni}}} + \frac{2}{I^2} \sum_{i < j} \frac{\partial G(P_i, P_j)}{\partial \sigma_{x_{ni}}} \right]. \quad (11)$$

The partial derivatives $\frac{\partial G(P_i, P_j)}{\partial \mu_{x_{ni}}}$, $\frac{\partial G(P_i, P_j)}{\partial \sigma_{x_{ni}}}$ and $\frac{\partial H(P_i)}{\partial \sigma_{x_{ni}}}$ are:

$$\frac{\partial G(P_i, P_j)}{\partial \mu_{x_{ni}}} = \frac{(\mu_{x_{nj}} - \mu_{x_{ni}})}{(\sigma_{x_{ni}} + \sigma_{x_{nj}})} G(P_i, P_j), \quad (12)$$

$$\frac{\partial G(P_i, P_j)}{\partial \sigma_{x_{ni}}} = [.] G(P_i, P_j), \text{ where,} \quad (13)$$

¹The complete derivation of analytic solution to integrals in functions $G(,)$ and $H()$ is provided in the accompanying supplementary material.

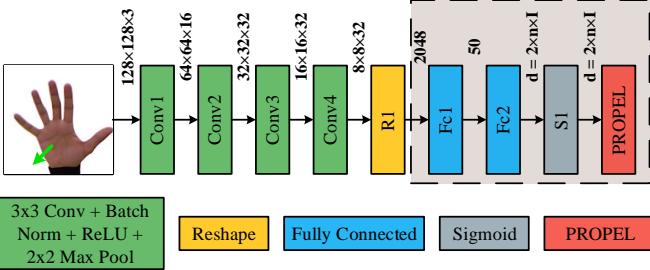


Fig. 3: Flowchart showing the CNN regression architecture used for experimental validation of the proposed PROPEL loss (details in Section IV-A). The green arrow (\uparrow) shows the target GT hand orientation that is only used during training. Scalar d represents the dimension of the network output. The comparison methods, described in Section IV-B, use the same overall architecture, where only the components in the highlighted box are replaced to match output dimensions required by a comparison loss function.

$$[.] = \frac{(\mu_{x_{ni}}^2 + \mu_{x_{nj}}^2 - 2\mu_{x_{ni}}\mu_{x_{nj}} - \sigma_{x_{ni}} - \sigma_{x_{nj}})}{2(\sigma_{x_{ni}} + \sigma_{x_{nj}})^2}, \quad (14)$$

$$\frac{\partial H(P_i)}{\partial \sigma_{x_{ni}}} = \frac{-1}{2\sigma_{x_{ni}}} H(P_i). \quad (15)$$

At each training iteration, PROPEL computes the loss in the forward pass using Equation 6 on the output from the CNN model. For the backward pass, the partial derivatives are used along with the GT labels to backpropagate the error and optimize the parameters using RMSProp [27].

IV. EXPERIMENTAL VALIDATION

We perform experimental validation of PROPEL by addressing image-based hand and head orientation regression problems. Given a dataset $\mathcal{U} = \{(\mathbf{C}_k, \mathbf{o}_k)\}_{k=1}^K$ with K uncalibrated color images \mathbf{C}_k of hands or heads, and the corresponding target orientation vectors \mathbf{o}_k , the orientation regression task involve learning the mapping of color images \mathbf{C}_k onto the target orientations \mathbf{o}_k . For hands, the orientation vector $\mathbf{o}_k = \{\phi_k, \psi_k\}^\top \in \mathbb{R}^2$ contains Azimuth (ϕ_k) and Elevation (ψ_k) angles, resulting from pronation/supination of the forearm and flexion/extension of the wrist, respectively [5]. Whereas head orientation is defined by yaw (ϕ_k), pitch (ψ_k) and roll (χ_k) angles, i.e. $\mathbf{o}_k = \{\phi_k, \psi_k, \chi_k\}^\top \in \mathbb{R}^3$ [7]. We note that the problem of hand orientation regression is specifically challenging as there may be similar hand shapes that map onto multiple orientations. We evaluate PROPEL on the hand orientation dataset from [5], which contains 9414 images collected from 22 participants. The range of hand orientation angles captured by this dataset are defined within a circular space with $\sqrt{\phi^2 + \psi^2} \leq 45^\circ$. For head orientation, we validate on publicly available BIWI dataset [7], which contains 10k images from 20 participants.

A. Network Architecture

The CNN network utilized for experimental validation of PROPEL is inspired from VGG networks [4] (Fig. 3). We keep

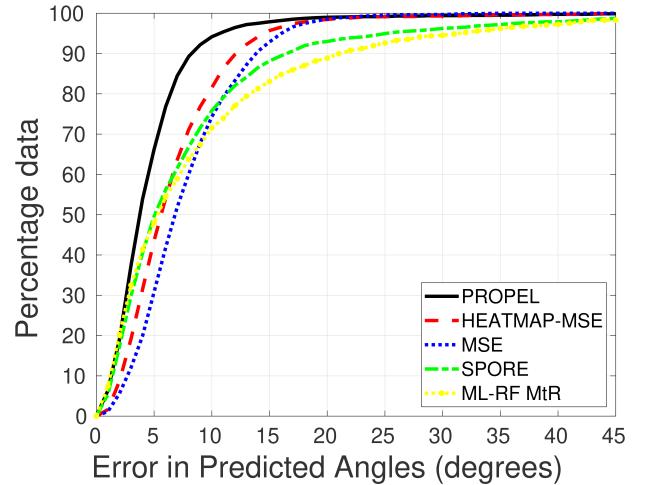


Fig. 4: Percentage data versus prediction error shows the percentage data that lies below an error threshold for single-fold validation.

the number of filters in each layer lower than in [4], as this does not have a significant impact on our model's performance. For hand orientation regression, the input to our network is an RGB color image \mathbf{C}_k of size $128 \times 128 \times 3$, where the method learns to infer parameters for the model distribution P_m in Equation 3. The maximum likelihood estimate (MLE) of the inferred parametric probability distribution is used to get predicted hand orientation angles. Our network has four 3×3 convolutional layers, where each layer is followed by batch normalization, a rectified non-linear unit (ReLU) and a 2×2 max pooling layer. The output from the convolutional layers is flattened and fed into two fully connected layers, Fc1 and Fc2 with 50 and $d = 2 \times n \times I$ neurons, respectively. In our model distribution P_m , each dimension n requires two parameters, i.e. the mean and the variance of a Gaussian. P_m may have I Gaussians resulting in $d = 2 \times n \times I$ parameters as output of Fc2.

The head orientation validation is done with the same network, however as the size of head images in the BIWI dataset is small, the input images have size 64×64 . We also modify Fc2 to enable inference of three-dimensional head orientation, i.e. $n = 3$.

B. Comparison Methods

The proposed method is compared with existing state-of-the-art methods for hand orientation regression, namely, Marginalization through Regression (MtR) and Staged PrObabilistic REgression (SPORE) [20, 5]. Both MtR and SPORE use hand-crafted features along with probabilistic Random Forest regressors. Utilizing mean squared error (MSE) loss as baseline, we show that a CNN model, without any probabilistic output, can significantly outperform existing methods that use hand-crafted features. As PROPEL addresses probabilistic regression for a CNN, we compare it with a method inspired from [13, 14] that learns probability heatmap using MSE loss

TABLE I: Mean Absolute Error (MAE) in degrees along with number of model parameters for single-fold and 5-fold cross-validation.

Single-fold Validation				
Method	Azimuth (ϕ)	Elevation (ϕ)	CMAE	No. of Parameters
PROPEL (proposed)	5.27°	3.99°	4.63	126,890
HEATMAP-MSE [13]	7.22°	6.07°	6.65	1,248,932
MSE	8.23°	7.18°	7.71	126,584
SPORE [5]	8.49°	7.26°	7.88	-
Mtr [20]	9.67°	7.97°	8.82	-
5-fold Cross-validation				
Method	Azimuth (ϕ)	Elevation (ϕ)	CMAE	No. of Parameters
PROPEL (proposed)	11.96°	10.00°	10.98	126,890
HEATMAP-MSE [13]	13.81°	10.42°	12.12	1,248,932
MSE	14.16°	11.51°	12.84	126,584
SPORE [5]	15.73°	12.95°	14.34	-
Mtr [20]	16.16°	12.83°	14.50	-

(HEATMAP-MSE). The heatmap distributions are generated by evaluating Equation 2 within a 20×20 grid covering the domain $\phi \in [-45^\circ, +45^\circ]$ and $\psi \in [-45^\circ, +45^\circ]$. Both MSE and HEATMAP-MSE use the same network architecture as PROPEL, except for the changes in the highlighted box in Fig. 3. MSE uses Fc2 with two neurons, whereas HEATMAP-MSE requires 500 neurons for Fc1 and 400 neurons for Fc2. The additional neurons are required to enable HEATMAP-MSE to learn $20 \times 20 = 400$ dimensional target heatmap distributions.

We also compare PROPEL on the image-based head orientation regression task with existing literature. Performance of PROPEL is compared to existing CNN methods [8, 9] and a hand-crafted feature-based method [23]. [9] also utilizes synthetic data, however we only include their results from real data for a fair comparison. Comparison with our MSE loss baseline is also made. We exclude HEATMAP-MSE from this comparison as it becomes unwieldy in 3D, and head orientation is represented in 3D.

MSE and HEATMAP-MSE are independently trained for experimental validation. Both PROPEL and HEATMAP-MSE require additional ground truth (GT) variances to be defined in Equation 2. A large variance can over-smooth the GT PDF, producing underfitted prediction PDFs. However, a small variance yields a GT PDF that captures unwanted variations, resulting in overfitted model PDFs. We empirically found that a variance of $(9^\circ)^2$, for all angles, can optimally capture the variations within the datasets, while enabling both PROPEL and HEATMAP-MSE to generalize well. Additionally PROPEL also requires the number of Gaussians I in the model PDF P_m . We found that $I = 2$ results in outperforming existing methods while showing the significance of having multiple components in P_m .

Following the approach in [5], we utilize Mean Absolute Error (MAE) and Combined Mean Absolute Error (CMAE) for evaluating the overall performance of all comparison methods.

C. Experimental Validation for Hand Orientation Regression

We perform single-fold experimental evaluation by randomly dividing the dataset into training (70%), testing (20%)

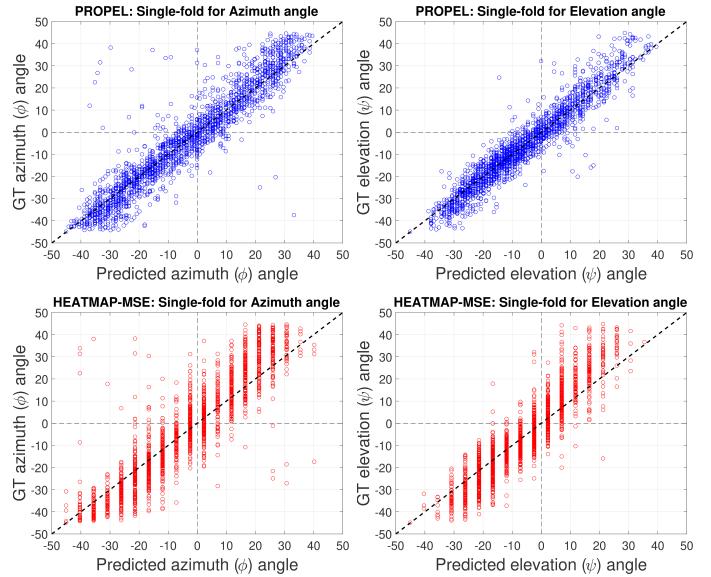


Fig. 5: GT versus predicted angle plot using PROPEL (in blue \circ) and HEATMAP-MSE (in red \circ) shows Azimuth (ϕ) angles (left column) and Elevation (ψ) angles (right column). A good regressor predicts angles close to GT angles, resulting in a diagonal line.

and validation (10%) sets. This evaluates the generalization performance of the comparison methods against a scenario where the training dataset is large enough to cover variations in hand shape, color, size and orientations. Additionally, we also evaluate the performance of our method using 5-fold cross-validation, where the folds are created by grouping multiple participants’ data.

Table I shows Mean Absolute Error (MAE) along with the number of model parameters for CNN methods in the single-fold validation. Further, in Fig. 1 (a) we show the accuracy and model parameters trade-off. We observe that the proposed PROPEL method outperforms existing state-of-the-art in both hand-crafted feature-based as well as CNN-based hand orientation regression. Probabilistic regression in both PROPEL and HEATMAP-MSE outperforms the widely used MSE loss. This is due to the ability of probabilistic methods to provide better generalization while addressing ambiguities in cases where multiple hand orientations can result in similar projected hand shape [5]. From Fig. 1 (a), we note that PROPEL requires 10x fewer model parameters as compared to HEATMAP-MSE as it enables a CNN to directly learn parameters for model PDF P_m . Furthermore, we note from Table I that Azimuth (ϕ) angles have greater MAE as compared to Elevation (ψ) angles for all comparison methods. This is due to the variations in inter-finger separation and styles for performing pronation/supination of the forearm across different participants in the hand orientation dataset. Fig. 4 provides further insight by showing the percentage of data that lies under a given error threshold for all comparison methods. It can be seen that with PROPEL 95% of the data

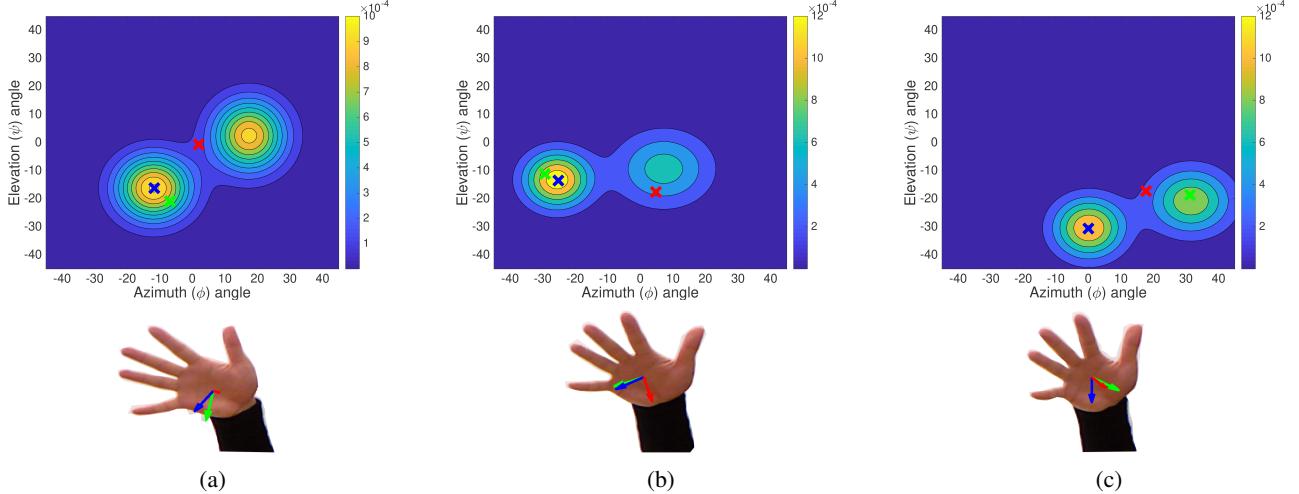


Fig. 6: Ability of PROPEL to handle ambiguity in predictions, showing input hand images and the corresponding predicted PDFs from PROPEL. Hand orientation angles from GT (in green ↑×), and predictions from PROPEL (in blue ↑×) and MSE (in red ↑×) are shown. (a) - (b) show cases where PROPEL’s PDFs successfully handles ambiguities, whereas (c) shows a failure case where PROPEL prediction contains a large error. Note that even in failure mode, PROPEL’s PDF contains useful information regarding the correct prediction.

lies below 10° error, while HEATMAP-MSE has an error of 15° for the same percentage data.

In Fig. 6, we show some of the ambiguous cases and compare the probabilistic output from PROPEL against predictions from MSE. PROPEL has the ability to learn multiple hypotheses, due to the presence of multiple parameterized Gaussians in the model PDF P_m . During training, the error is minimized for ambiguous targets where these Gaussians are able to model multiple hypotheses instead of being forced to a single hypothesis (as in MSE). On the other hand, MSE prediction *tries* to fit into the target space (see Fig 6 (a)-(b)). Fig. 6 (c) shows a failure case where PROPEL prediction results in a large error. We observe that even in such a case, the predicted model PDF P_m contains useful information regarding the correct prediction.

The most closely related method to PROPEL is HEATMAP-MSE as it also provides probabilistic regression by learning probability heatmaps. Table I, Fig. 1 (a) and Fig. 5 show a comparison of these methods. We make two observations from this comparison. First, PROPEL uses approximately $10 \times$ fewer model parameters as compared to HEATMAP-MSE, while providing significantly better accuracy. Secondly, from Fig. 5 (second row), quantization due to 20×20 heatmaps becomes the main source of errors in HEATMAP-MSE. Furthermore, it should be noted that with higher dimensional targets HEATMAP-MSE will require an exponentially larger number of model parameters, whereas our proposed PROPEL method will only result in increasing n which is linearly related by $d = 2 \times n \times I$ to the model parameters.

The 5-fold cross-validation helps in understanding the generalization capability of each comparison method for inferring hand orientation for unseen participants’ hand shape, color and size. Table I shows results from this experiment. Once again,

TABLE II: MAE for experimental validation on BIWI dataset (*indicates the use of evaluation protocol from [8])

Method	Pitch	Yaw	Roll	CMAE	No. of Parameters
Unseen Faces (training=21 users and testing=3 users)*					
PROPEL (proposed)	3.44°	4.02°	3.28°	3.58°	50, 294
MSE	9.20°	6.30°	5.15°	6.88°	49, 835
Lathuilière et al. [8]	4.68°	3.12°	3.07°	3.62°	135, 847, 232
Liu et al. [9]	6.10°	6.00°	5.94°	6.01°	595, 573
Drouard et al. [23]	5.43°	4.24°	4.13°	4.60°	-
Unseen Faces (Leave-one-out validation)					
PROPEL (proposed)	5.93°	5.81°	4.53°	5.42°	50, 294
MSE	7.70°	6.70°	5.81°	6.74°	49, 835
Liu et al. [9]	6.00°	6.10°	5.70°	5.17°	595, 573
Drouard et al. [23]	5.90°	4.90°	4.70°	5.93°	-

PROPEL outperforms all comparison methods by achieving lowest MAE, which shows that utilizing PROPEL results in a CNN model that is able to generalize better than the existing state-of-the-art for CNN regression loss.

D. Experimental Validation for Head Orientation Regression

For this comparison, we use the established experimental protocols from [8] and leave-one-out validation [23]. Table II shows comparison of PROPEL with existing state-of-the-art methods. Note that PROPEL is trained without any data augmentation, and the model is learned end-to-end unlike [8] that uses a pre-trained CNN. In contrast, PROPEL can learn to infer probabilities for any higher dimensional target. PROPEL achieves competitive results with other methods and outperforms all techniques (including the state-of-the-art) in one of three angles, in both evaluation protocols. These experiments also demonstrate the generalization of PROPEL to higher dimensional problems, for which existing heatmap-based probabilistic regression methods require exponentially more model parameters. We further visualize accuracy and

model parameter trade-off in Fig. 1 (b) and note that PROPEL significantly reduces the number of required model parameters, while achieving similar accuracy as the state-of-the-art method from [8]. These results indicate the potential for models trained with PROPEL, which may need much less parameters to achieve close to state-of-the-art accuracy.

V. LIMITATIONS AND FUTURE WORK

PROPEL represents a probability distribution as a superposition of a finite number of Gaussians. In this paper, two Gaussians are used, based on prior knowledge that there may be two possible solutions to hand or head orientation as a result of the symmetry problem. Using a finite number of Gaussians is a strength in this paper as doing so constrains and regularises the solution space; however, for other problems which require more complex target distributions, we note a higher (possibly infinite) number of Gaussians may be required.

There are quite a few future directions for this work that can be explored. We plan to explore application of PROPEL to higher dimensional regression problems such as joint location in the hand or human body where having unconstrained probabilistic regression can improve the state-of-the-art. Another interesting aspect of PROPEL that can be studied is to include additional covariance terms in the model distributions to exploit dependence within target labels.

VI. CONCLUSION

We proposed a novel PROBabilistic Parametric rEgression Loss (PROPEL) which enabled learning parametric probabilities for addressing regression problems using CNN. PROPEL is fully differentiable with an analytic closed-form solution to integrals, which allows it to be used with existing CNN architectures. A complete generalizable derivation for different level of complexity of prediction probabilities and multi-variate target dimensions was provided using Mixture of Gaussians. Comprehensive experimental validation showed that PROPEL outperforms previous state-of-the-art. It shows better generalization capabilities while reducing the number of model parameters by $10\times$. We also showed the usefulness of parametric probabilities for ambiguous cases where PROPEL handled predictions by providing multiple hypotheses. Our contribution can enable CNN regression models to have better prediction capabilities for addressing a range of problems.

REFERENCES

- [1] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.
- [2] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in *Advances in neural information processing systems*, 2012, pp. 1097–1105.
- [3] E. Shelhamer, J. Long, and T. Darrell, “Fully convolutional networks for semantic segmentation,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 4, pp. 640–651, 2017.
- [4] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” in *International Conference on Learning Representations*, 2015.
- [5] M. Asad and G. Slabaugh, “Spore: Staged probabilistic regression for hand orientation inference,” *Computer Vision and Image Understanding*, 2017.
- [6] S. R. Fanello, C. Keskin, S. Izadi, P. Kohli, D. Kim, D. Sweeney, A. Criminisi, J. Shotton, S. B. Kang, and T. Paek, “Learning to be a depth camera for close-range human capture and interaction,” *ACM Transactions on Graphics (TOG)*, vol. 33, no. 4, p. 86, 2014.
- [7] G. Fanelli, M. Dantone, J. Gall, A. Fossati, and L. Van Gool, “Random forests for real time 3d face analysis,” *International Journal of Computer Vision*, vol. 101, no. 3, pp. 437–458, 2013.
- [8] S. Lathuiliere, R. Juge, P. Mesejo, R. Munoz-Salinas, and R. Horaud, “Deep mixture of linear inverse regressions applied to head-pose estimation,” in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 3, 2017, p. 7.
- [9] X. Liu, W. Liang, Y. Wang, S. Li, and M. Pei, “3d head pose estimation with convolutional neural network trained on synthetic images,” in *Image Processing (ICIP), 2016 IEEE International Conference on*. IEEE, 2016, pp. 1289–1293.
- [10] D. Eigen, C. Puhrsch, and R. Fergus, “Depth map prediction from a single image using a multi-scale deep network,” in *Advances in neural information processing systems*, 2014, pp. 2366–2374.
- [11] S. Lathuilière, P. Mesejo, X. Alameda-Pineda, and R. Horaud, “A comprehensive analysis of deep regression,” *IEEE transactions on pattern analysis and machine intelligence*, 2019.
- [12] A. Bulat and G. Tzimiropoulos, “Human pose estimation via convolutional part heatmap regression,” in *European Conference on Computer Vision*. Springer, 2016, pp. 717–732.
- [13] L. Ge, H. Liang, J. Yuan, and D. Thalmann, “Robust 3d hand pose estimation in single depth images: from single-view cnn to multi-view cnns,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 3593–3601.
- [14] J. Tompson, M. Stein, Y. Lecun, and K. Perlin, “Real-time continuous pose recovery of human hands using convolutional networks,” *ACM Transactions on Graphics (ToG)*, vol. 33, no. 5, p. 169, 2014.
- [15] S. M. M. R. Al Arif, K. Knapp, and G. Slabaugh, “Probabilistic spatial regression using a deep fully convolutional neural network,” in *British Machine Vision Conference*. The British Machine Vision Association, 2017.
- [16] T. Pfister, J. Charles, and A. Zisserman, “Flowing convnets for human pose estimation in videos,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1913–1921.
- [17] G. Moon, J. Yong Chang, and K. Mu Lee, “V2v-posesnet: Voxel-to-voxel prediction network for accurate 3d hand and human pose estimation from a single depth map,” in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.
- [18] C. Payer, D. Stern, H. Bischof, and M. Urschler, “Integrating spatial configuration into heatmap regression based cnns for landmark localization,” *Medical Image Analysis*, vol. 54, pp. 207–219, 2019.
- [19] D. Pathak, P. Krähenbühl, S. X. Yu, and T. Darrell, “Constrained structured regression with convolutional neural networks,” *arXiv preprint arXiv:1511.07497*, 2015.
- [20] M. Asad and G. Slabaugh, “Learning marginalization through regression for hand orientation inference,” in *Computer Vision and Pattern Recognition (CVPR) Second Workshop on Observing and Understanding Hands in Action (HANDS)*, 2016.
- [21] ———, “Hand orientation regression using random forest for augmented reality,” in *International Conference on Augmented and Virtual Reality*, 2014.
- [22] L. Yang, Z. Qi, Z. Liu, H. Liu, M. Ling, L. Shi, and X. Liu, “An embedded implementation of cnn-based hand detection and orientation estimation algorithm,” *Machine Vision and Applications*, vol. 30, no. 6, pp. 1071–1082, 2019.
- [23] V. Drouard *et al.*, “Head pose estimation via probabilistic high-dimensional regression,” in *ICIP*, 2015.
- [24] B. Ahn, J. Park, and I. S. Kweon, “Real-time head orientation from a monocular camera using deep neural network,” in *Asian Conference on Computer Vision*. Springer, 2014, pp. 82–96.
- [25] G. Sfikas, C. Constantinopoulos, A. Likas, and N. Galatsanos, “An analytic distance metric for gaussian mixture models with application in image retrieval,” *Artificial Neural Networks: Formal Models and Their Applications—ICANN 2005*, pp. 755–755, 2005.
- [26] J. R. Hershey *et al.*, “Approximating the kullback leibler divergence between gaussian mixture models,” in *ICASSP*, 2007.
- [27] T. Tieleman and G. Hinton, “Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude,” *COURSERA: Neural networks for machine learning*, vol. 4, no. 2, pp. 26–31, 2012.

Supplementary material for PROPEL: Probabilistic Parametric Regression Loss for Convolutional Neural Networks

Muhammad Asad
Imagination Technologies
Kings Langley, UK
masadc@gmail.com

Rilwan Basaru
Onaria Technologies
London, UK
remi@onariatech.com

S M Masudur Rahman Al Arif
ASML
Veldhoven, Netherlands
masudur.al.arif@asml.com

Greg Slabaugh
City, University of London
London, UK
greg.slabaugh@gmail.com

This supplementary document presents additional results and analytic solution to equations in the main paper.

I. ADDITIONAL QUALITATIVE RESULTS

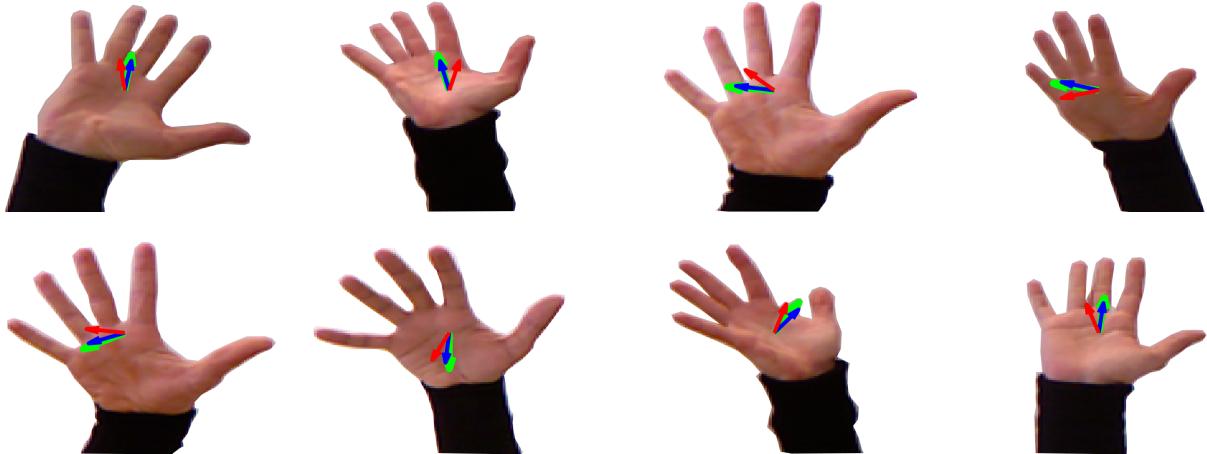


Fig. 1: Qualitative results showing input hand images, GT and predicted orientations from PROPEL and HEATMAP-MSE. Hand orientation angles from GT (in green ↑), and predictions from PROPEL (in blue ↑) and HEATMAP-MSE (in red ↑) are shown. Note that predictions from PROPEL are more accurate, while HEATMAP-MSE always contains an error with respect to GT orientations. The error in HEATMAP-MSE is due to quantization of heatmaps which is limited by the number of model parameters required, whereas PROPEL utilizes 10× lesser parameters while generating continuous distributions that are free from quantization errors.

II. DERIVATION OF ANALYTIC SOLUTION TO INTEGRALS

This section includes derivations of analytic solution to integrals for function $G(\cdot)$ and $H(\cdot)$ presented in the Section 3.2 of the paper. Let $\mathbf{x} = \{x_1, x_2, \dots, x_n\}^\top \in \mathbb{R}^n$ define the target prediction space with n dimensions. In order to facilitate analytic solution of our proposed loss function, we write the function $G(P_i, P_j)$ between two Gaussian distributions P_i and P_j as:

$$G(P_i, P_j) = \int P_i P_j d\mathbf{x} = \int \cdots \int P_i P_j dx_1 \cdots dx_n, \quad (1)$$

$$= \int \cdots \int \left[\frac{e^{-\frac{1}{2} \left[\frac{(x_1 - \mu_{x_{1i}})^2}{\sigma_{x_{1i}}} + \dots + \frac{(x_n - \mu_{x_{ni}})^2}{\sigma_{x_{ni}}} \right]}}{(\sqrt{2\pi})^n \sqrt{\sigma_{x_{1i}} \cdots \sigma_{x_{ni}}}} \right] \left[\frac{e^{-\frac{1}{2} \left[\frac{(x_1 - \mu_{x_{1j}})^2}{\sigma_{x_{1j}}} + \dots + \frac{(x_n - \mu_{x_{nj}})^2}{\sigma_{x_{nj}}} \right]}}{(\sqrt{2\pi})^n \sqrt{\sigma_{x_{1j}} \cdots \sigma_{x_{nj}}}} \right] dx_1 \cdots dx_n, \quad (2)$$

$$= \int \cdots \int \frac{e^{-\frac{1}{2} \left[\frac{(x_1 - \mu_{x_{1i}})^2}{\sigma_{x_{1i}}} + \frac{(x_1 - \mu_{x_{1j}})^2}{\sigma_{x_{1j}}} + \dots + \frac{(x_n - \mu_{x_{ni}})^2}{\sigma_{x_{ni}}} + \frac{(x_n - \mu_{x_{nj}})^2}{\sigma_{x_{nj}}} \right]}}{(2\pi)^n \sqrt{(\sigma_{x_{1i}} \cdots \sigma_{x_{ni}})(\sigma_{x_{1j}} \cdots \sigma_{x_{nj}})}} dx_1 \cdots dx_n, \quad (3)$$

$$= \int \cdots \int \frac{e^{-\frac{1}{2} \left[\frac{(x_1^2 - 2x_1\mu_{x_{1i}} + \mu_{x_{1i}}^2)\sigma_{x_{1j}} + (x_1^2 - 2x_1\mu_{x_{1j}} + \mu_{x_{1j}}^2)\sigma_{x_{1i}}}{\sigma_{x_{1i}}\sigma_{x_{1j}}} + \dots + \frac{(x_n - \mu_{x_{ni}})^2}{\sigma_{x_{ni}}} + \frac{(x_n - \mu_{x_{nj}})^2}{\sigma_{x_{nj}}} \right]}}{(2\pi)^n \sqrt{(\sigma_{x_{1i}} \cdots \sigma_{x_{ni}})(\sigma_{x_{1j}} \cdots \sigma_{x_{nj}})}} dx_1 \cdots dx_n, \quad (4)$$

$$= \int \cdots \int \frac{e^{-\frac{1}{2} \left[-\frac{\sigma_{x_{1i}} + \sigma_{x_{1j}}}{2\sigma_{x_{1i}}\sigma_{x_{1j}}} x^2 + \frac{\mu_{x_{1i}}\sigma_{x_{1j}} + \mu_{x_{1j}}\sigma_{x_{1i}}}{\sigma_{x_{1i}}\sigma_{x_{1j}}} x - \frac{1}{2} \left[\frac{\mu_{x_{1i}}^2\sigma_{x_{1j}} + \mu_{x_{1j}}^2\sigma_{x_{1i}}}{\sigma_{x_{1i}}\sigma_{x_{1j}}} + \dots + \frac{(x_n - \mu_{x_{ni}})^2}{\sigma_{x_{ni}}} + \frac{(x_n - \mu_{x_{nj}})^2}{\sigma_{x_{nj}}} \right] \right]}}{(2\pi)^n \sqrt{(\sigma_{x_{1i}} \cdots \sigma_{x_{ni}})(\sigma_{x_{1j}} \cdots \sigma_{x_{nj}})}} dx_1 \cdots dx_n. \quad (5)$$

Given $G(P_i, P_j)$, we can now analytically evaluate the Gaussian for $\int [\cdot] dx_1$ using the following theorem ¹:

$$\int_{-\infty}^{+\infty} e^{-ax^2 + bx - c} dx_1 = e^{\frac{b^2}{4a} - c} \sqrt{\frac{\pi}{a}}. \quad (6)$$

We compare left hand side of Equation 6 with Equation 5 to determine a , b and c as:

$$a = \frac{\sigma_{x_{1i}} + \sigma_{x_{1j}}}{2\sigma_{x_{1i}}\sigma_{x_{1j}}}, \quad (7)$$

$$b = \frac{\mu_{x_{1i}}\sigma_{x_{1j}} + \mu_{x_{1j}}\sigma_{x_{1i}}}{\sigma_{x_{1i}}\sigma_{x_{1j}}}, \quad (8)$$

$$c = \left[\frac{\mu_{x_{1i}}^2\sigma_{x_{1j}} + \mu_{x_{1j}}^2\sigma_{x_{1i}}}{2\sigma_{x_{1i}}\sigma_{x_{1j}}} + \dots + \frac{(x_n - \mu_{x_{ni}})^2}{2\sigma_{x_{ni}}} + \frac{(x_n - \mu_{x_{nj}})^2}{2\sigma_{x_{nj}}} \right]. \quad (9)$$

Replacing a , b and c in the right hand side of Equation 6 to get Equation:

$$G(P_i, P_j) = \int \cdots \int \frac{e^{-\frac{1}{2} \left[\frac{2\mu_{x_{1i}}\mu_{x_{1j}} - \mu_{x_{1i}}^2 - \mu_{x_{1j}}^2}{2(\sigma_{x_{1i}} + \sigma_{x_{1j}})} - \frac{1}{2} \left[\frac{(x_2 - \mu_{x_{2i}})^2}{\sigma_{x_{2i}}} + \frac{(x_2 - \mu_{x_{2j}})^2}{\sigma_{x_{2j}}} + \dots + \frac{(x_n - \mu_{x_{ni}})^2}{\sigma_{x_{ni}}} + \frac{(x_n - \mu_{x_{nj}})^2}{\sigma_{x_{nj}}} \right] \right]}}{(2\pi)^{n-1} \sqrt{2\pi(\sigma_{x_{1i}} + \sigma_{x_{1j}})(\sigma_{x_{2i}}\sigma_{x_{2j}} \cdots \sigma_{x_{ni}}\sigma_{x_{nj}})}} dx_2 \cdots dx_n. \quad (10)$$

We can apply similar analytic solution for $\int \cdots \int [\cdot] dx_2 \cdots dx_n$ to arrive at the Equation:

$$G(P_i, P_j) = \frac{e^{\left[\frac{2\mu_{x_{1i}}\mu_{x_{1j}} - \mu_{x_{1i}}^2 - \mu_{x_{1j}}^2}{2(\sigma_{x_{1i}} + \sigma_{x_{1j}})} + \dots + \frac{2\mu_{x_{ni}}\mu_{x_{nj}} - \mu_{x_{ni}}^2 - \mu_{x_{nj}}^2}{2(\sigma_{x_{ni}} + \sigma_{x_{nj}})} \right]}}{(\sqrt{2\pi})^n \sqrt{(\sigma_{x_{1i}} + \sigma_{x_{1j}}) \cdots (\sigma_{x_{ni}} + \sigma_{x_{nj}})}}. \quad (11)$$

¹Gaussian Integral (accessed on 10-03-2020) <http://mathworld.wolfram.com/GaussianIntegral.html>

The second function we evaluate for helping our derivation is $H(P_1)$ defined as:

$$H(P_i) = \int P_i^2 d\underline{x} = \int \cdots \int P_i^2 dx_1 \cdots dx_n, \quad (12)$$

$$= \int \cdots \int \left(\frac{e^{-\frac{1}{2} \left[\frac{(x_1 - \mu_{x_{1i}})^2}{\sigma_{x_{1i}}} + \cdots + \frac{(x_n - \mu_{x_{ni}})^2}{\sigma_{x_{ni}}} \right]}}{(\sqrt{2\pi})^n \sqrt{\sigma_{x_{1i}} \cdots \sigma_{x_{ni}}}} \right)^2 dx_1 \cdots dx_n, \quad (13)$$

$$= \int \cdots \int \frac{e^{-\left[\frac{(x_1 - \mu_{x_{1i}})^2}{\sigma_{x_{1i}}} + \cdots + \frac{(x_n - \mu_{x_{ni}})^2}{\sigma_{x_{ni}}} \right]}}{(2\pi)^n \sigma_{x_{1i}} \cdots \sigma_{x_{ni}}} dx_1 \cdots dx_n, \quad (14)$$

$$= \int \cdots \int \frac{e^{-\left[\frac{x_1^2 - 2x_1\mu_{x_{1i}} + \mu_{x_{1i}}^2}{\sigma_{x_{1i}}} + \cdots + \frac{x_n^2 - 2x_n\mu_{x_{ni}} + \mu_{x_{ni}}^2}{\sigma_{x_{ni}}} \right]}}{(2\pi)^n \sigma_{x_{1i}} \cdots \sigma_{x_{ni}}} dx_1 \cdots dx_n. \quad (15)$$

Applying Equation 6, we reduce Equation 15 to:

$$H(P_i) = \frac{1}{(2\sqrt{\pi})^n \sqrt{\sigma_{x_{1i}} \cdots \sigma_{x_{ni}}}}. \quad (16)$$