Supplementary materials for CVAM-Pose: Conditional Variational Autoencoder for Multi-Object Monocular Pose Estimation

Jianyu Zhao
jzhao12@uclan.ac.uk
Wei Quan
wquan@uclan.ac.uk
Bogdan J. Matuszewski
bmatuszewski1@uclan.ac.uk

Computer Vision and Machine Learning (CVML) Group,
The University of Central Lancashire,
Preston, UK

We provide additional supplementary materials including:

1. Further quantitative and qualitative analyses of our method on the BOP version of the Linemod-Occluded dataset [1, 2, 6, 7].

2. More information on network implementations.

1 Additional Results on Linemod-Occluded

1.1 Quantitative Results

Pose Regression vs. LUT We conduct further ablation tests comparing the pose regression strategy used in our method to the lookup table (LUT) technique described in [13, 14]. The LUT technique assigns the rotation and projective distance from the most similar instance to the test instance, and utilises the centre of the bounding box as the 2D projective centre. This approach may lead to inaccuracies, particularly with heavily occluded objects or imprecise bounding boxes. In our analysis, the results for 3D rotation are reported using the AR_{MSPD} metric [6], while results for projective centre and distance are evaluated using the mean absolute error (MAE) metric. The choice of MAE over AR_{MSPD} is due to its parameter-free nature, which simplifies the interpretation of translational errors, as opposed to AR_{MSPD} that depends on predefined thresholds as outlined in [6].

Rotation	AR_{MSPD} ↑
LUT	0.666
Ours	**0.714**

Centre	MAE_{pixel} ↓
LUT	4.064
Ours	**2.913**

Distance	MAE_{mm} ↓
LUT	60.981
Ours	**43.278**

Table 1: Comparison between LUT and our regression method for the estimation of 3D rotation, 2D projective centre, and 2D projective distance.

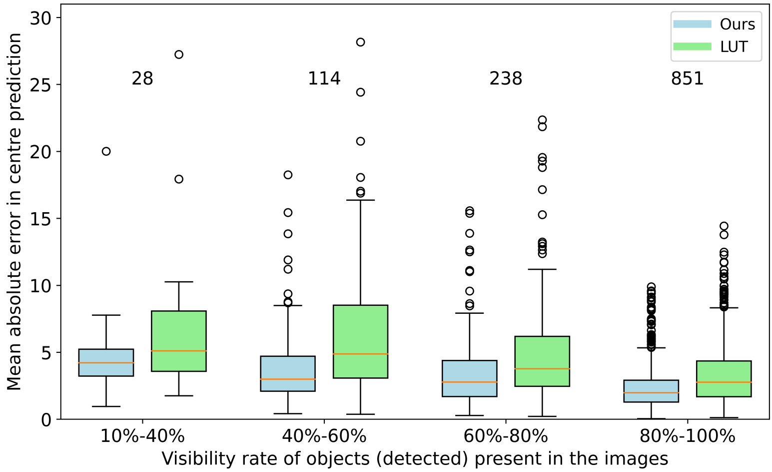

Figure 1: Box plots of the $\mathrm{MAE}_{\mathrm{pixel}}$ metric as a function of the objects' visibility rates. The number of data instances for each rate is shown above each pair of boxes.

As shown in Table 1, our continuous pose regression strategy demonstrates better results than using the LUT technique in estimating 3D rotation, 2D projective centre, and 2D projective distance, e.g. our method achieves smaller errors in distance measurement (improved by approximately 2% when computed in relation to the average object's distance in the test set). This can be attributed to the avoidance of the pose discretisation problem inherent in the LUT technique, particularly when the training data do not cover the entire $\mathrm{SO}(3)$. The performance of centre prediction is further illustrated in Fig. 1, which presents box plots quantifying the distribution of errors ($\mathrm{MAE}_{\mathrm{pixel}}$). It is evident that the median error in our method is consistently lower than that produced by the LUT technique across various visibility rates. The LUT method can also generate noticeable outlier errors in centre prediction, as high as 27 pixels.

Results on Individual Objects We also present additional results on individual objects from the Linemod-Occluded dataset [1, 2] in Table 2. The average recall of a single object, $\mathrm{AR}_{\mathrm{object}}$, is calculated from the average recall across the three metrics, $\mathrm{AR}_{\mathrm{VSD}}$, $\mathrm{AR}_{\mathrm{MSSD}}$, and $\mathrm{AR}_{\mathrm{MSPD}}$ [6]. The average value, denoted as **Avg.**, shows the main results for the entire dataset as already reported in the paper.

Object	ape	can	cat	driller	duck	eggbox	glue	holepuncher	**Avg.**
$\mathrm{AR}_{\mathrm{VSD}}$	0.332	0.409	0.300	0.375	0.443	0.168	0.324	0.425	0.346
$\mathrm{AR}_{\mathrm{MSSD}}$	0.360	0.471	0.286	0.490	0.397	0.084	0.356	0.455	0.362
$\mathrm{AR}_{\mathrm{MSPD}}$	0.830	0.681	0.826	0.571	0.794	0.488	0.760	0.764	0.714
$\mathrm{AR}_{\mathrm{object}}$	0.507	0.520	0.471	0.479	0.545	0.247	0.480	0.548	0.475

Table 2: Results on the individual objects of the Linemod-Occluded dataset.

Among the three evaluation metrics, the MSPD metric demonstrates considerably higher accuracy than the others (25% higher on average). As explained in [6], this might be that the MSPD metric does not account for alignment along the optical axis, which is significant when evaluating on perspective images.

In terms of individual objects, the eggbox object exhibits lower accuracy than others (approximately 20% in $\mathrm{AR}_{\mathrm{object}}$), which might be associated with object symmetries, i.e. the pose ambiguity problem. To improve pose accuracy, especially for symmetrical objects, our method could be extended to estimate the distribution of potential poses through random sampling in the latent space, thereby better accommodating variances induced by object symmetries.

1.2 Qualitative Results

Fig. 2 visualises pose estimation results on two randomly selected images from the Linemod-Occluded dataset, with poses estimated using CVAM-Pose. The target objects, including ape, cat, driller, duck, eggbox, glue, holepuncher, and iron, are rendered based on the estimated poses and reprojected onto the original test images. Correct estimations are represented by aligned reprojection masks, e.g. the cat object in the first image, while misaligned masks indicate incorrect estimations, e.g. the eggbox object in the first image.

2 Implementation Details

Network Architecture The proposed label-embedded CVAE network employs an adapted ResNet-18 [4] as the encoder, and a sequence of convolutional layers as the decoder. The ReLU activation function [11] is replaced with SiLU [3] to avoid the zero-gradient problem. The label-embedded MLP network consists of a series of fully connected layers with neurons $[256, 128, 64, 32, 16, out]$. Each hidden layer uses the SiLU activation and concatenates the one-hot encoded categorical labels with the output of the previous layer. The final output, out, varies depending on the regression task, such as 6 neurons for regressing the continuous 6D rotation representation [17].

Data Preprocessing The data preparation involves a crop-and-resize strategy proposed in [16]. This strategy crops images of the target objects into a square shape from the scene image using the ground truth bounding box, with the square's size defined by the longer side of the box. The cropped images of objects are resized to $128 \times 128 \times 3$ using bicubic interpolation, which matches the input size of the proposed CVAE network. Images, where less than 10% of the object's area is visible, are excluded, based on the visibility criteria defined in [6, 15]. Approximately $40k$ images per object are obtained, with 90% designated for training and the remaining 10% for validation. For test data preparation, the crop-and-resize strategy is also applied, using the detection bounding boxes provided by a pre-trained Mask-RCNN detector [5, 8].

Training Parameters All experiments are implemented in PyTorch [12]. The label-embedded CVAE and MLP networks are trained using the AdamW optimiser [9] with parameters set as follows: $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\varepsilon = 1e-8$, and $\lambda = 0.01$. The initial learning rate is set to $1e-4$ for CVAE and $3e-3$ for MLPs, with scheduled reductions by a factor of 0.2 when the validation loss does not improve over a "patience" period (50 epochs for CVAE, 500 for MLPs). Training terminates when the lowest learning rate of $1e-6$ is reached, and no improvement in validation loss occurs for N epochs ($N = 50$ for CVAE and $N = 1000$

Figure 2: Example visualisation of the estimated poses using CVAM-Pose. The rendering process uses the Pyrender software [10]. The images of objects are taken from the BOP version of the Linemod-Occluded dataset [1, 2, 6, 7].

for MLPs). The CVAE network is trained with a batch size of 128, while MLPs process all inputs per batch. For reproducibility, all random seeds are fixed at 0.

References

[1] Eric Brachmann. 6d object pose estimation using 3d object coordinates [data], 2020. URL https://doi.org/10.11588/data/V4MUMX.

[2] Eric Brachmann, Alexander Krull, Frank Michel, Stefan Gumhold, Jamie Shotton, and Carsten Rother. Learning 6d object pose estimation using 3d object coordinates. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part II 13*, pages 536–551. Springer, 2014.

[3] Stefan Elfwing, Eiji Uchibe, and Kenji Doya. Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. *Neural networks*, 107:3–11, 2018.

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[5] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

[6] Tomas Hodan, Martin Sundermeyer, Bertram Drost, Yann Labbe, Eric Brachmann, Frank Michel, Carsten Rother, and Jiri Matas. Bop challenge 2020 on 6d object localization. In *Computer Vision–ECCV 2020 Workshops: Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*, pages 577–594. Springer, 2020.

[7] Tomas Hodan, Martin Sundermeyer, Bertram Drost, Yann Labbe, Eric Brachmann, Frank Michel, Carsten Rother, and Jiri Matas. Datasets., 2020. URL https://bop.felk.cvut.cz/datasets/.

[8] Yann Labbé, Justin Carpentier, Mathieu Aubry, and Josef Sivic. Cosypose: Consistent multi-view multi-object 6d pose estimation. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVII 16*, pages 574–591. Springer, 2020.

[9] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Bkg6RiCqY7.

[10] Matthew Matl. Pyrender: Easy-to-use gltf 2.0-compliant opengl renderer for visualization of 3d scenes., 2018. URL https://github.com/mmatl/pyrender.

[11] Vinod Nair and Geoffrey E Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pages 807–814, 2010.

[12] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

[13] Martin Sundermeyer, Maximilian Durner, En Yen Puang, Zoltan-Csaba Marton, Narunas Vaskevicius, Kai O Arras, and Rudolph Triebel. Multi-path learning for object pose estimation across domains. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13916–13925, 2020.

[14] Martin Sundermeyer, Zoltan-Csaba Marton, Maximilian Durner, and Rudolph Triebel. Augmented autoencoders: Implicit 3d orientation learning for 6d object detection. *International Journal of Computer Vision*, 128:714–729, 2020.

[15] Martin Sundermeyer, Tomáš Hodaň, Yann Labbe, Gu Wang, Eric Brachmann, Bertram Drost, Carsten Rother, and Jiří Matas. Bop challenge 2022 on detection, segmentation and pose estimation of specific rigid objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2784–2793, 2023.

[16] Jianyu Zhao, Edward Sanderson, and Bogdan J Matuszewski. Cvml-pose: Convolutional vae based multi-level network for object 3d pose estimation. *IEEE Access*, 11: 13830–13845, 2023.

[17] Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. On the continuity of rotation representations in neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5745–5753, 2019.