Deep Active Cross-Modal Visuo-Tactile Transfer Learning for Robotic Object Recognition

Prajval Kumar Murali, Cong Wang, Dongheui Lee, Ravinder Dahiya, and Mohsen Kaboli*

***Abstract*—We propose for the first time, a novel deep active visuo-tactile cross-modal full-fledged framework for object recognition by autonomous robotic systems. Our proposed network *xAVTNet* is actively trained with labelled point clouds from a vision sensor with one robot and tested with an active tactile perception strategy to recognise objects never touched before using another robot. We propose a novel visuo-tactile loss (*VTLoss*) to minimise the discrepancy between the visual and tactile domains for unsupervised domain adaptation. Our framework leverages the strengths of deep neural networks for cross-modal recognition along with active perception and active learning strategies for increased efficiency by minimising redundant data collection. Our method is extensively evaluated on a real robotic system and compared against baselines and other state-of-art approaches. We demonstrate clear outperformance in recognition accuracy compared to the state-of-art visuo-tactile cross-modal recognition method.**

***Index Terms*—Transfer Learning; Visuo-Tactile Cross-Modal Learning; Active Visuo-Tactile Object Recognition; Perception for Grasping and Manipulation**

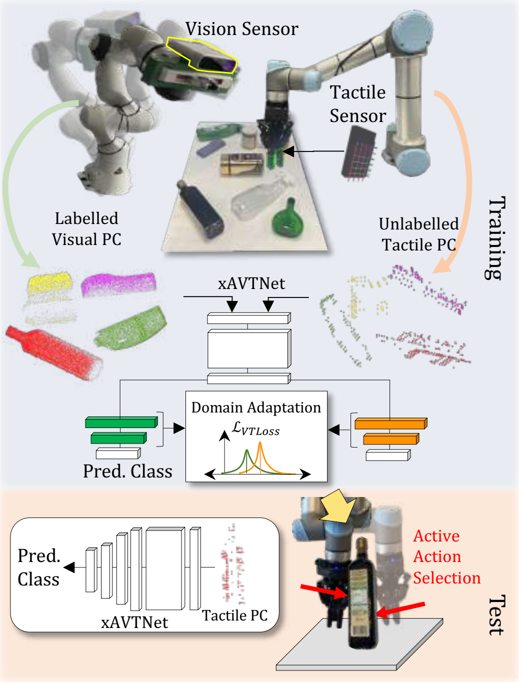


Fig. 1: Experimental setup: A Franka Emika Panda robot with a RGB-D vision sensor on the end-effector for active visual learning. A UR5 robot with 3-axis tactile sensor array on the gripper for deep cross-modal visuo-tactile transfer learning and active tactile object recognition.

I. INTRODUCTION

HUMANS from infants to adults can seamlessly transfer the knowledge gained from visual modality to the tactile modality in order to perceive and interact with objects in the environment especially during lack of visual feedback [1], [2]. For instance, we can identify and distinguish previously *seen* objects blindly only through *touch*. The human sensing and perception systems are also active such that the sensory systems are purposefully controlled to increase the information gained for the task at hand [3]. In [4]–[6], the researchers enabled robotics systems with the sense of touch to recognize object during in hand object or whole body manipulation. In this work, we aim to provide similar abilities to autonomous robots for cross-modal object recognition by actively training using visual modality and transferring to tactile modality without explicit training with the tactile modality as shown in Figure 1. This can provide increased autonomy and resilience for robots in unstructured environments. If visual sensing is unavailable due to various reasons such as occlusions, limited field of view, change in light intensity, dust blocking the sensor and so on, the robot is capable of completing the object recognition task using the tactile modality by leveraging only the previously gained knowledge from vision [7]–[9]. Furthermore, training an object recognition model with tactile sensing is time consuming due to sparsity of tactile data, human annotation and need for interaction with objects whereas through cross-modal learning, the robot can exploit the *a priori* gained knowledge using visual sensing to recognise objects during *testing* stage through only tactile sensing. Moreover, through active tactile perception and learning, the robot can autonomously reduce the number of actions to perceive objects physical properties and to learn efficiently about objects and discriminate them among each other. [10]–[13].

Visuo-tactile cross-modal perception and learning is a challenging problem due to the weak-pairing between visual and

Manuscript received: February 24, 2022; Revised May 28, 2022; Accepted June 27, 2022.

This paper was recommended for publication by Editor Markus Vincze upon evaluation of the Reviewers' comments. This work was supported in part by BMW Group and the European Commission via INTUITIVE (Grant agreement ID: 861166)

* Corresponding author: Mohsen Kaboli, mohsen.kaboli@bmwgroup.com P.K. Murali, C. Wang, and M.Kaboli are with the BMW Group, RoboTac Lab, München, Germany. e-mail: name.surname@bmwgroup.com

P.K. Murali and R. Dahiya are with the University of Glasgow, Scotland, e-mail: 2584615m@student.gla.ac.uk, Ravinder.Dahiya@glasgow.ac.uk

C. Wang is with the Technical University of Munich, Germany, e-mail: ge57qom@mytum.de

D. Lee is with the TU Wien, Austria, e-mail: dongheui.lee@tuwien.ac.at

M. Kaboli is with the Donders Institute for Brain and Cognition, Radboud University, Netherlands, e-mail: mohsen.kaboli@donders.ru.nl

Digital Object Identifier (DOI): see top of this page.

tactile data: (a) variation in density of information from each modality, (b) scale gap as vision sensors can capture the global scene while tactile sensors capture local object geometry, (c) temporal misalignment as vision sensors capture data in one-shot while tactile sensors capture data sequentially, and (d) tactile data are inherently action conditioned as data depends on the type of action that is performed [14], [15]. Few works in literature address the problem of visuo-tactile cross-modal learning. Falco et al. [16] tackled the problem for vision-to-tactile object recognition through shape. In [16], visual and tactile features from point clouds are extracted using a hand-crafted feature descripter termed termed CLUE (Cross modal point cLoUd dEscriptor) and Geodesic flow kernel (GFK) [17] was used for domain adaptation. For other applications, Zapata-Impata et al. [18] tackled the problem of grasp stability estimation and proposed a vision-to-tactile cross modal learning approach to generate tactile data from input visual data for predicting stability of grasping prior to contact with the object. Similarly, Yunzhu et al. [19] addressed the problem of synthesizing plausible tactile signals from visual inputs and predicted visual images given tactile input using a condition generative adversarial network in the 2D image domain using RGB images sequences from GelSight tactile sensor and camera. Kaboli et al. [11], [20] proposed for the first time an active tactile transfer learning approach which leveraged prior tactile knowledge to effectively discriminate novel objects using tactile sensing.

As we focus on vision-to-tactile cross-modal object recognition through shape, we aim to train our network models with dense point clouds that are generally accessible from visual sensors and employ directly on sparse point clouds measured with tactile sensors during testing. Recently point clouds-based approaches using deep learning have received research interest as various types of 3D acquisition devices such as LiDARs, RGB-D sensors and 3D scanners are becoming increasingly available. Deep learning methods on point clouds are challenging due to the unstructured nature of point clouds, high dimensionality and relatively small-scale datasets [21]. The seminal works PointNet [22] and PointNet++ [23] were proposed to work directly on raw 3D point clouds for tasks such as object classification and semantic segmentation. However, the performance of such networks drops significantly in the case of sparse point clouds with point numbers ranging from 10-100 [24]. Such sparse point clouds are typical from LiDAR data and tactile sensing [25]–[27]. Retraining deep neural networks on sparse tactile data is prohibitively expensive due to temporal costs of tactile data collection and annotation. Fortunately with unsupervised domain adaptation (UDA), the richly labelled visual (source) data can be leveraged to minimise the domain shift with the unlabelled tactile (target) dataset. While there are various ways for UDA available in literature [28], [29], we focus on discrepancy based techniques such as maximum mean discrepancy (MMD) [30] and correlation alignment (CORAL) [31] which aim to reduce the distance between the source and target domains using statistic criteria. Furthermore, contrasting to visual perception, tactile-based recognition requires interaction with the objects as data is collected upon contact with the objects [8]. To reduce redundant data collection, temporal costs and human intervention, several approaches have been proposed for performing active data acquisition through information-gain based action selection [10]–[12], [32]–[34]. Leveraging active perception and learning techniques can aid in reducing data collection costs and improve time efficiency for vision-to-tactile cross-modal domain adaptation.

Our contributions are as follows:

(I) We propose a novel framework for deep active visuo-tactile cross-modal robotic object recognition. Our deep neural network (termed *xAVTNet*) is trained solely with dense visual point cloud data and tested on sparse point clouds acquired from tactile sensors.

(II) We propose a novel unsupervised domain adaptation loss function termed *VTLoss* for minimising the domain gap between the visual and tactile domain.

(III) We propose an *active* deep learning framework for visual object learning for reducing redundant data collection and annotation. Furthermore, we propose an *active* tactile-based object recognition approach to reduce the number of tactile actions.

(IV) We perform extensive robotic experiments to show the validity of our approach and compare with state-of-art method.

II. METHODS

A. Problem Description

We propose a novel framework shown in Figure 2 for the task of deep active visuo-tactile cross-modal object recognition. Our proposed network termed *xAVTNet* (cross active visuo-tactile network) is trained with labelled source domain dataset $D_s = \{(\mathbf{x}_i^s, y_i^s)\}_{i=1}^{n_s}$ with n_s samples from vision domain constructed using an active learning strategy by querying uncertain samples from a larger unlabelled dataset D_u consisting of n_u samples with $n_u \gg n_s$ (Figure 2(a)). Given the labelled source domain $D_s = \{(\mathbf{x}_i^s, y_i^s)\}_{i=1}^{n_s}$ from vision domain and unlabelled target domain $D_t = \{\mathbf{x}_j^t\}_{j=1}^{n_t}$ with n_t samples from tactile domain, the model is adapted by reducing the domain discrepancy through our proposed *VTLoss* (Figure 2(b)). The adapted model is used for active tactile-based object recognition wherein the robot is tasked to reason upon possible tactile touch actions to perform and chooses the next best touch which maximises the expected information gain (Figure 2(c)).

B. Deep Active Visual Object Learning

Network Architecture: Our network takes as input the unordered point cloud representing one object consisting of m points where each point is a vector of (x,y,z) coordinates. It outputs k probabilistic classification scores for all k candidate classes. We use PointNet [22] as the backbone for feature extraction. PointNet applies input and feature transformations and aggregates the point features by max pooling to a global feature vector of size 1024 [22]. The global feature vector is followed by three fully-connected (fc) layers of size $512, 256, k$. We denote the mapping from input point clouds to output classes as G_v and associated parameters by θ. The

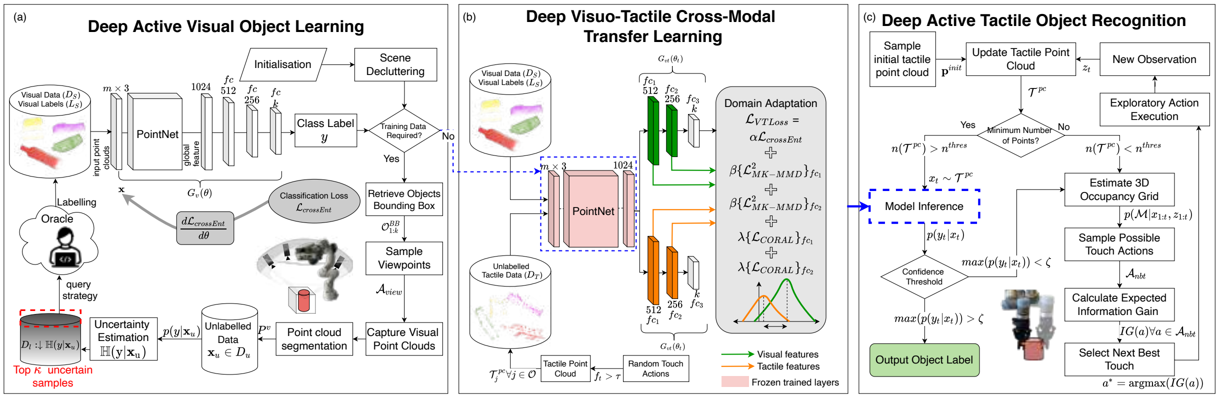

(a)
Deep Active Visual Object Learning
Initialisation
Scene Decluttering
Visual Data (D_S)
Visual Labels (L_S)
$m \times 3$
input point clouds
PointNet
global feature
1024
fc 512
fc 256
fc k
Class Label y
Training Data Required?
No
Yes
$G_v(\theta)$
$\mathbf{x}$
Classification Loss $\mathcal{L}_{crossEnt}$
$\frac{d\mathcal{L}_{crossEnt}}{d\theta}$
Retrieve Objects Bounding Box
$\mathcal{O}^{BB}_{1:k}$
Sample Viewpoints
$\mathcal{A}_{view}$
Capture Visual Point Clouds
Point cloud segmentation
P^v
Unlabelled Data $\mathbf{x}_u \in D_u$
$p(y|\mathbf{x}_u)$
Uncertainty Estimation $\mathbb{H}(\mathrm{y}|\mathbf{x}_\mathrm{u})$
$D_l :\downarrow \mathbb{H}(y|\mathbf{x}_u)$
Top κ uncertain samples
query strategy
Oracle
Labelling
(b)
Deep Visuo-Tactile Cross-Modal Transfer Learning
Visual Data (D_S)
Visual Labels (L_S)
Unlabelled Tactile Data (D_T)
$m \times 3$
PointNet
1024
$G_{vt}(\theta_t)$
fc_1 512
fc_2 256
fc_3 k
Domain Adaptation
$\mathcal{L}_{VTLoss}$ =
$\alpha\mathcal{L}_{crossEnt}$
+
$\beta\{\mathcal{L}^2_{MK-MMD}\}_{fc_1}$
+
$\beta\{\mathcal{L}^2_{MK-MMD}\}_{fc_2}$
+
$\lambda\{\mathcal{L}_{CORAL}\}_{fc_1}$
+
$\lambda\{\mathcal{L}_{CORAL}\}_{fc_2}$
$\mathcal{T}^{pc}_j \forall j \in \mathcal{O}$
Tactile Point Cloud
$f_t > \tau$
Random Touch Actions
Visual features
Tactile features
Frozen trained layers
(c)
Deep Active Tactile Object Recognition
Sample initial tactile point cloud
$\mathbf{p}^{init}$
Update Tactile Point Cloud
z_t
New Observation
Exploratory Action Execution
$\mathcal{T}^{pc}$
Minimum Number of Points?
Yes
No
$n(\mathcal{T}^{pc}) > n^{thres}$
$n(\mathcal{T}^{pc}) < n^{thres}$
$x_t \sim \mathcal{T}^{pc}$
Model Inference
$p(y_t|x_t)$
Confidence Threshold
$max(p(y_t|x_t)) < \zeta$
$max(p(y_t|x_t)) > \zeta$
Output Object Label
Estimate 3D Occupancy Grid
$p(\mathcal{M}|x_{1:t}, z_{1:t})$
Sample Possible Touch Actions
$\mathcal{A}_{nbt}$
Calculate Expected Information Gain
$IG(a)\forall a \in \mathcal{A}_{nbt}$
Select Next Best Touch
$a^* = \mathrm{argmax}(IG(a))$

Fig. 2: Proposed framework for deep active visuo-tactile cross modal object recognition. Figure (a) describes the proposed active learning method for training our deep network *xAVTNet* with visual point cloud data, (b) describes the proposed unsupervised domain adaptation of *xAVTNet* with unlabelled tactile data and (c) describes the active tactile object recognition using the cross-modal adapted model.

xAVTNet is trained with dense point clouds from vision sensor of the real world objects with $k = 12$ classes. The visual point clouds are subsampled to 1024 points before passing to *xAVTNet*. We use the cross-entropy loss for training. The visual point cloud dataset D_s representing the source domain is collected using an active learning technique as detailed below.

Visual Viewpoint Sampling and Visual Data Collection: In order to collect visual training data of the objects, we use a vision sensor attached to a robot capable of choosing arbitrary viewpoints in 3D space limited by the workspace and kinematic constraints of the robot. Choosing different viewpoints of the same objects helps to improve the predictive robustness of the network as the same object can appear differently based on the view. While commanding the robot to arbitrary viewpoints, it is crucial to maintain the viewing angle of the camera such that the object of interest lies within its field of view (FoV). A viewpoint $a^{view} \in \mathcal{A}^{view}$ is defined as the 3D position $\mathbf{p}^{view} \in \mathbb{R}^3$ and orientation $\mathbf{R}^{view} \in SO(3)$ of the camera frame. We perform Markov Monte-Carlo sampling of N viewpoints on the hemisphere space located above the centroid $\mathbf{o}_{centroid}$ of the bounding box of the object of interest which is known *a priori*. The 3D position $\mathbf{p}^{view}$ is randomly sampled as a point on the hemisphere and the orientation of the view as axis of rotation $\vec{\mathbf{e}}$ and angle θ is computed with [34]:

$$\vec{\mathbf{h}} = \frac{\mathbf{p}^{view} - \mathbf{o}_{centroid}}{||\mathbf{p}^{view} - \mathbf{o}_{centroid}||}, \tag{1}$$

$$\theta = \cos^{-1}(\vec{\mathbf{h}} \cdot \vec{\mathbf{Z}}), \quad \vec{\mathbf{e}} = \frac{\vec{\mathbf{h}} \times \vec{\mathbf{Z}}}{||\vec{\mathbf{h}} \times \vec{\mathbf{Z}}||}, \tag{2}$$

where $\vec{\mathbf{Z}} = \{0,0,1\}$ is the Z-axis of the world frame. Using the resulting angle-axis formulation $(\vec{\mathbf{e}}, \theta)$ from (2), we can derive the equivalent rotation matrix $\mathbf{R}^{view}$ using the Rodrigues formula. $\mathbf{R}^{view}$ ensures that the camera is always oriented towards the object of interest. The robot is commanded to N viewpoints sequentially and the point clouds are extracted from each viewpoint. The process is repeated for different object in the scene and the objects in the scene are also rearranged periodically by the human in order to ensure that all faces of the object is captured by the robot. The raw point clouds corresponding to each object are processed in order to remove outlier noisy points as well as the base plane and added to the unlabelled dataset $\mathbf{x}_u \in D_u$.

Uncertainty Estimation and Query Strategy: The goal of active learning is to select the samples from the unlabelled dataset D_u, which upon labelling and training improves the model accuracy significantly with fewer training samples. In order to select such samples from the unlabelled dataset, we use the predictive probability of the network $p(y|\mathbf{x}_u)$ to determine uncertainty. The softmax function provides the predictive probability of an input sample. However as noted by prior works [35], [36], the softmax function may provide inconsistent predictions as it gives higher probability to unseen data. Hence, we adopt the Monte Carlo dropout (MC-dropout) method instead to extract the uncertainty [37]. The MC-dropout technique [37] casts dropout training in deep neural networks as approximate Bayesian inference in deep Gaussian processes. It works by performing multiple stochastic feed-forward passes through the network with dropout active at test time and averaging the results. In particular, it is defined as

$$p(y|\mathbf{x}_u) = \frac{1}{T}\sum_{t=1}^{T} p(y|\mathbf{x}_u, \mathbf{W}_t), \tag{3}$$

where $\mathbf{W}_t$ refers to the weights of the network at the t^{th} inference and T refers to the total number of stochastic forward-passes. Given the predictive probability, we can quantify the uncertainty of the samples by measuring the Shannon Entropy as:

$$\mathbb{H}(y|\mathbf{x}_u) = -\sum_{c=1}^{k} p(y = c|\mathbf{x}_u) \log p(y = c|\mathbf{x}_u), \tag{4}$$

here $c = 1, 2, \ldots k$ refers to number of distinct objects used in our experiments. We order the unlabelled dataset based on the Shannon entropy values and query the top κ samples into dataset D_l as the most informative samples for labelling. The samples in D_l are labelled by the oracle (human annotator) and added to the training dataset for further training. The procedure is repeated until a stopping condition is satisfied such as the performance of the network does not improve over successive iterations or a desired accuracy is reached.

C. Deep Visuo-Tactile Cross-Modal Object Learning

The challenge of domain adaptation arises from the fact that the target domain (tactile modality) has no labelled data, hence fine-tuning our trained network on the source domain to the target domain directly is impossible. Another challenges stems from the density and sparsity of visual and tactile data respectively. We use the dense visual data of the order of 1024 points while the sparse tactile data usually contains 30-80 points. We exploit the available labelled source data and the unlabelled target data to minimise the discrepancy of the distributions of the two domains in the hidden representations of the fully connected layers as shown in Figure 2(b). We utilise discrepancy-based methods to extract domain-invariant representations for the unsupervised domain adaptation. Popular techniques in literature among discrepancy-based methods include Maximum Mean Discrepancy (MMD) [38] and Correlation Alignment (CORAL) [31]. Given labelled source domain $D_s = \{(x_i^s, y_i^s)\}_{i=1}^{n_s}$ with n_s samples and unlabelled target domain $D_t = \{x_j^t\}_{j=1}^{n_t}$ with n_t samples which are represented by probability distributions p^s and p^t respectively, MMD between p^s and p^t is defined as:

$$\text{MMD}^2(p^s, p^t) = \sup_{||\phi||_{\mathcal{H}} \leq 1} ||\mathbb{E}_{x^s \sim p^s}[\phi(x^s)] - \mathbb{E}_{x^t \sim p^t}[\phi(x^t)]||^2_{\mathcal{H}}, \tag{5}$$

where $\mathcal{H}$ is the reproducing kernel Hilbert space (RKHS), $\phi(\cdot)$ is the feature mapping associated with the kernel map $k(x^s, x^t) =< \phi(x^s), \phi(x^t) >$, $\sup(\cdot)$ is the supremum of the input aggregate and $||\phi||_{\mathcal{H}} \leq 1$ defines a set of functions in the unit ball of $\mathcal{H}$. We utilise the multi-kernel MMD (MK-MMD) [39] which assumes that the optimal kernel is obtained by the linear combination of many kernels. Herein the kernel $k(x^s, x^t)$ is defined as the convex combination of b positive semi-definite kernels $\{k_u\}$ [39]:

$$K \triangleq \left\{ k = \sum_{u=1}^{b} \beta_u k_u : \sum_{u=1}^{b} \beta_u = 1, \beta_u \geq 0, \forall u \right\} \tag{6}$$

where k is weighted by different kernel and the coefficients β_u is the weight to ensure that the generated multi-kernel k is characteristic. In contrast to MK-MMD which compares all order of statistics, CORAL [31] is another discrepancy measure which attempts to align the second-order statistics of the source and target distributions. Deep-CORAL [40] extends CORAL for deep neural networks and is defined as follows:

$$\text{CORAL}(x^s, x^t) = \frac{1}{4d^2} ||C_s - C_t||^2_F, \tag{7}$$

where $||.||^2_F$ is the squared matrix Frobenius norm, C_s and C_t are the covariance matrices of the source and target domain data. We propose to combine MK-MMD, CORAL and the supervised classification loss as a *weighted linear combination* to devise the loss function. As reported by Sun et al. [40], MMD applies symmetric transformations whereas CORAL applies asymmetric transformations to the source and target domain. Intuitively, symmetric transformations attempt to extract a subspace that neglects the dissimilarities between the two domains whereas asymmetric transformations attempt to "bridge" the different domains [40]. By combining MK-MMD, CORAL and the classification loss, it can provide a good trade-off in minimising the discrepancies between the domains and help in improving the UDA performance as shown empirically in Section III. We freeze the PointNet layers that provide the 1024 feature vector during domain adaptation training and only train the fc layers. The PointNet layers are capable of learning domain independent features that can be transferred to the target domain whereas the fc are tailored to the original task on the source domain and require domain adaptation training for the target domain. We denote the neural network layer mapping consisting of fc_1, fc_2 and fc_3 layers as $G_{vt}(\theta_t)$. We perform *multi-layer domain adaptation* with fully-connected layers fc_1 and fc_2 as we empirically found to achieve higher target domain accuracy in comparison to single layer adaptation with fc_1 or fc_2. It has been shown in prior works [41], that adapting a single layer does not sufficiently undo the dataset bias between the source and target domains due to the other non-transferable fc layers. Hence, our proposed $\mathcal{L}_{VTLoss}$ is defined as:

$$\begin{aligned}\mathcal{L}_{VTLoss} = \alpha \mathcal{L}_{crossEnt} + \beta\{\mathcal{L}^2_{MK-MMD}\}_{fc_1} + \beta\{\mathcal{L}^2_{MK-MMD}\}_{fc_2} \\ + \lambda\{\mathcal{L}_{CORAL}\}_{fc_1} + \lambda\{\mathcal{L}_{CORAL}\}_{fc_2},\end{aligned} \tag{8}$$

where α, β, λ are hyperparameters. The domain adaptation network architecture of *xAVTNet* is shown in Figure 2(b). The discrepancy between the source and target domain is reduced by minimising the $\mathcal{L}_{VTLoss}$ as $\min_{G_{vt}(\theta_t)} \mathcal{L}_{VTLoss}$.

D. Deep Active Tactile Object Recognition

Given the trained network *xAVTNet* using source domain visual data and unsupervised domain adaptation with unlabelled tactile data, the objective during test stage is to classify the object with only tactile data that is collected actively by maximising the expected information gain. We define a tactile action $\mathbf{a}$ as a ray represented by a tuple $\mathbf{a} = (\mathbf{s}, \overrightarrow{\mathbf{d}})$, with $\mathbf{s}$ as the start point and $\overrightarrow{\mathbf{d}}$ the direction of the ray. We assume the 3D bounding box pose of the object is given. The actions are performed as guarded motions so that the robot does not accidentally push or topple the object. We discretize the 3D bounding box into a 3D occupancy grid $\mathcal{OG}$ with resolution g_{res}. Each cell c_i in the occupancy grid is represented by a Bernoulli random variable and has an occupancy probability $p(c_i)$. There are two possible states for each cell with $c_i = 1$ indicating the cell is occupied and $c_i = 0$ for an empty cell. A common independence assumption of each cell with other cells enables the calculation of the overall entropy of the occupancy grid as the summation of the entropy of each cell. The Shannon Entropy of the entire grid can be computed as [42]:

$$\mathbb{H}(\mathcal{OG}) = -\sum_{c_i \in \mathcal{OG}} p(c_i) log(p(c_i)) + (1 - p(c_i)) log(1 - p(c_i)) \tag{9}$$

To compute the next best touch (NBT), we compute the expected entropy-based information gain. As it is intractable to calculate the exact entropy from a predicted touch, we perform a common simplifying approximation by predicting the expected measurements $\hat{z}_t$ from an action $\mathbf{a}_t$ at time t using ray-traversal algorithms. A virtual sensor model is defined

TABLE I: The number of labelled samples required to reach a certain relative accuracy measured by the relative error to the fully train network

	No. of Labelled Samples			
Relative error	10%	5%	3%	2%
Random strategy (baseline)	2000	4500	5000	5500
Active strategy (ours)	2000	2000	3000	3500

TABLE II: Ablation study with the domain adaptation methods

Domain Adaptation	**V → T Accuracy (%)**
MMD	57.28 ± 0.77
CORAL	58.34 ± 1.11
VTLoss$_{fc1}$	70.42 ± 1.23
VTLoss$_{fc2}$	62.19 ± 2.48
VTLoss	**81.25** ± 1.97

TABLE III: Confusion matrix for tactile object recognition

1	0.85	0	0	0	0	0	0	0.05	0.1	0	0	0
2	0.2	0.75	0	0	0	0	0.05	0	0	0	0	0
3	0	0	0.95	0	0	0.05	0	0	0	0	0	0
4	0	0	0	1	0	0	0	0	0	0	0	0
5	0.1	0	0	0.05	0.8	0	0.05	0	0	0	0	0
6	0	0	0	0	0	1	0	0	0	0	0	0
7	0	0	0	0	0	0	0.9	0	0	0	0	0.1
8	0.7	0	0	0	0	0	0	0.2	0	0	0.1	0
9	0.05	0	0	0	0	0	0	0	0.95	0	0	0
10	0.05	0	0	0	0	0	0	0.1	0	0.45	0.4	0
11	0	0	0	0	0	0	0	0	0	0	1	0
12	0	0	0	0	0.05	0	0	0	0.05	0	0	0.9
Object	1	2	3	4	5	6	7	8	9	10	11	12

TABLE IV: Comparison study with state-of-art approach

Method	**V → T Accuracy (%)**
CLUE+GFK+1NN [16]	26.72 ± 0.96
CLUE+GFK+SVM [16]	15.05 ± 2.54
xAVTnet-noDA	51.25 ± 2.25
xAVTNet (ours)	**81.25** ± 1.97

representing the tactile sensor with n_{tax} taxels casting a set of rays $\mathcal{R} = \{r_1, r_2, \dots r_{n_{tax}}\}$ for a given distance d_{ray} in the *z-axis* of the sensor model coordinate frame, with one ray per taxel. We perform Monte-Carlo sampling of N_{nbt} possible touch points from possible actions $\mathcal{A}_{nbt}$ on each face of the bounding box except the bottom face as the object rests on a flat surface. The grid cells which are traversed by the rays are computed to be occupied or free and the respective log-odds is updated accordingly [43]:

$$l(\hat{z}^{view}) = \begin{cases} log\frac{p_h}{1-p_h} & \text{if } \hat{z}^{view} \hat{=}\ hit \\ log\frac{p_m}{1-p_m} & \text{if } \hat{z}^{view} \hat{=}\ miss \end{cases} \tag{10}$$

where p_h and p_m are the probabilities of hit and miss which are user-defined values set to 0.7 and 0.4 respectively as in [43]. Given the expected observations from all the possible touch points and the updated probabilities of each grid cell, we can evaluate the expected entropy of the overall grid by Equation (9). The expected information gain by taking a touch action $\mathbf{a}_t$ and corresponding expected measurement $\hat{z}_t$ is given by the Kullback–Leibler (KL) divergence between the posterior entropy after integrating the expected measurements and the prior entropy [44]:

$$E[\mathbb{I}(p(c_i|\mathbf{a}_t, \hat{z}_t))] = \mathbb{H}(p(c_i)) - \mathbb{H}(p(c_i|\mathbf{a}_t, \hat{z}_t)) \tag{11}$$

Hence, the selected action $\mathbf{a}_t^{nbt*}$ is given by:

$$\mathbf{a}_t^{nbt*} = \underset{\mathbf{a} \in \mathcal{A}}{\arg\max}(E[\mathbb{I}(p(c_i|\mathbf{a}_t, \hat{z}_t))]) \tag{12}$$

As shown in the Figure 2(c), the object classification procedure is started with an initial set of tactile points $\mathbf{p}^{init}$ that is acquired by performing random tactile touch actions. The random tactile touch actions are sampled randomly on the bounding box of the object and performed using guarded motions as explained above. After each action, the acquired points are collated into the tactile point cloud $\mathcal{T}^{pc}$. A minimum number of points in the tactile point cloud (N_{min}^T) are required to perform model inference. If the current number of points in the tactile point cloud $n(\mathcal{T}^{pc})$ is less than N_{min}^T and/or if the output confidence from the model inference is less than a threshold ζ, then active touch actions are performed in order to acquire additional touch points ${}^t\mathbf{p}$.

III. EXPERIMENTS

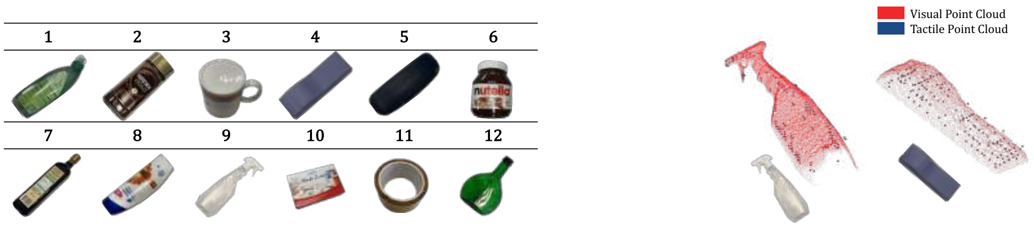
A. Experimental Setup and Data Collection

The experimental setup shown in Figure 1 consists of a Universal Robots UR5 robot with a Robotiq 2F140 Gripper and a Franka Emika Panda robot with the standard Panda Gripper. The Robotiq 2F140 fingertips are equipped with tactile sensor arrays from XELA Robotics[1]. The tactile sensing system consists of 140 taxels that provide 3-axis force measurements on each taxel in the sensor coordinate frame. We normalise the raw values received from the sensor. We use the outer finger (24 taxels) and the finger tip (6 taxels) in order to acquire the tactile data. We use straight line trajectories with guarded motions to collect the data. When the force value measured on any of the taxels exceeds the threshold $f_r > \tau_f$ (set to 1.1), the motion is stopped and the 3D locations of the excited taxels are recorded as the tactile point cloud ${}^tS$. The tactile point cloud is expressed in the common world coordinate frame $\mathcal{W}$ using the robot's kinematic model. An Azure Kinect DK RGB-D camera is rigidly attached to the Panda Gripper with a custom designed flange which provides the vision point cloud ${}^v\mathcal{S}$ and is expressed in the world-frame using hand-eye calibration [45]. Both visual and tactile point clouds are only composed of the x, y, z coordinates and other properties such as normals, colour are not used. The camera can also be used to extract the bounding box pose of the object using point cloud segmentation and clustering methods from the Point Cloud Library [2]. We also used the OctoMap library [3] for the next best touch implementations. We used a ROS-based framework for controlling the robots, sensor acquisitions and data collection. **Network Implementation:** The PointNet [22] layers performing input and feature transformations to encode a 1024 global feature vector is used. It is followed by three fully connected layers fc_1,fc_2,fc_3 of size 512, 256 and k respectively. The hidden layers fc_1, fc_2 include ReLU

[1] https://xelarobotics.com/en/home

[2] https://pointclouds.org/

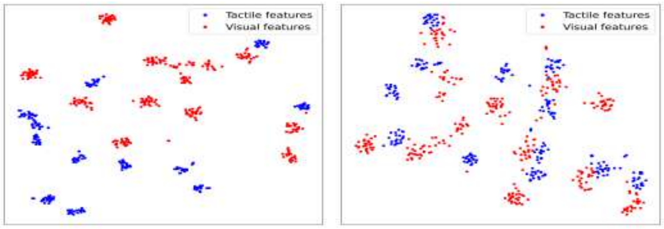
[3] https://octomap.github.io/



(a)

(b)

Fig. 3: (a) Experimental objects: Twelve daily objects with different characteristic properties such as shape and transparency selected for object recognition task (b) Vision and tactile point clouds of select objects shown overlapped to demonstrate the difference in point densities.



(a) Before domain adaptation

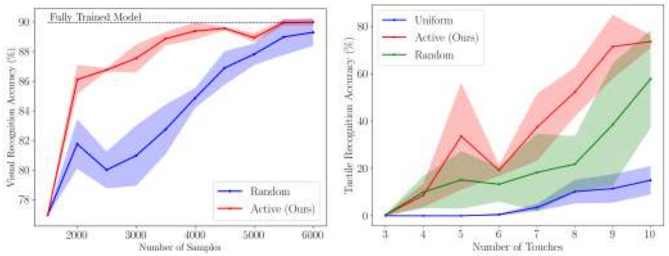
(b) After domain adaptation

Fig. 4: (a) Visual and tactile features before domain adaptation (b) after performing domain adaptation.

and batch normalisation. Furthermore, we set Dropout with probability 0.4 on the fc_2 layer. We used ADAM optimiser and learning rate set to 10^{-3}. Two streams of fc layers are used for domain adaptation as shown in Figure 2(b). We use the hyper-parameters that are empirically tuned for our method: $\alpha = 10$, $\beta = 10$ and $\lambda = 10$. The robot experiments were performed on a workstation running Ubuntu 18.04 with 8 core Intel i7-8550U CPU @ 1.80GHz and 16 GB RAM. The training and domain adaptation of the network was performed using PyTorch framework on a workstation with NVidia Quadro RTX 4000 GPU with 8 GB RAM. We use a set of 12 objects for the task of object recognition as shown in Figure 3(a). The objects are selected based on varying degree of shape complexity and transparency that is challenging for visual sensors. We also show the visual and tactile point clouds of some objects in Figure 3(b) highlighting the difference in the number of points and point density between the two domains. Although we use dynamic viewpoints for the visual point cloud acquisition, some regions of the objects remain occluded due to the kinematic limits of the fixed-base manipulator. A point cloud from a viewpoint is considered as one training sample and we do not merge point clouds from various views. The visual sensor may also produce noisy measurements and warped point clouds due to acute viewing angles which have been retained to make our network robust to real-world sensors.

B. Robot Experiments

Deep Active Visual Object Learning: If the scene is cluttered, we use our prior work in Murali et al. [34] to autonomously declutter the workspace. After the scene is decluttered, the robots initiate visual data collection. We collected a total of 9300 visual point clouds for the 12 objects by autonomously commanding the Panda robot to different viewpoints. The total dataset includes the data augmentation performed by random rotations around the Z-axis to be rotation invariant. The scene is also manually rearranged between data collection iterations in order to capture all possible views of the object. However, it should be noted that our model is not affected by the relative pose of the objects. We used 6000 samples for training, 1500 for validation and 1800 as test set. All the training samples are unlabelled and represent the unlabelled dataset D_u. We randomly select 1500 samples from D_u and label them using a human annotator and train our network. The trained network is used to compute the uncertainty of the remaining unlabelled samples as explained in Section II-B. We compare our active learning strategy with a baseline method that randomly queries samples from D_u. At each query step, $\kappa = 500$ samples are queried from the unlabelled dataset. We present the mean (solid line) and standard deviation (shaded region) results for deep active visual object learning versus baseline comparison in Figure 5(a). We also report the number of labelled samples necessary to achieve a certain relative error of the fully trained network in Table I.



(a) Vision: Active learning

(b) Tactile: Active recognition

Fig. 5: (a) Active strategy versus random strategy for deep visual learning (solid line: mean, shaded: standard deviation). (b) Active strategy versus uniform and random strategy for tactile object recognition (solid line: median, shaded: median absolute deviation).

Visuo-Tactile Domain Adaptation: We collected 10 tactile point clouds for each object using random tactile collection strategy. Similar to the visual dataset, we augment our tactile dataset by performing random rotations around the Z-axis in order to be rotation invariant increasing the dataset to 100 point clouds per object. All the tactile point clouds are unlabelled as our objective is to perform unsupervised domain adaptation. We used 900 samples for domain adaptation and 300 samples as test set. Table III shows the confusion matrix for the classification accuracy in the test set after performing unsupervised domain adaptation using our proposed *VTLoss* function. In order to show the performance of domain adaptation method, we also compare against the MMD loss and CORAL loss as ablation studies shown in Table II. Since we also performed *multi-layer* domain adaptation, we studied the variants wherein a single hidden layer $fc1$ or $fc2$ is used for domain adaptation and the results are reported in Table II.

To benchmark our proposed framework, we compare against the visuo-tactile cross-modal domain adaptation work of Falco et al. [16] which is closely related to our work. Due to unavailability of official source code from their work, we have re-implemented the paper as follows: we implemented the CLUE descriptor using PCL and the geodesic flow kernel (GFK) for domain adaptation using MATLAB domain adaptation toolbox[4]. We used k-nearest neighbours (kNN) classifier and support vector machines (SVM) for classification and the parameters have been fine-tuned according to guidelines in [16]. We must note that our re-implementation may not be identical to that of the original implementation. We used 3D objects of complex shapes compared to quasi-planar objects in their work. Furthermore, we directly employ the dense visual and sparse tactile point clouds without equalizing the point clouds from the two domains. We report the test classification accuracy with tactile data in Table IV.

Deep Active Tactile Object Recognition: Our proposed method is independent of the method of tactile data collection. The data can be recorded uniformly, randomly or even actively exploiting information gain. In order to evaluate our active tactile recognition method, we compare against the uniform and random collection strategy. The uniform collection strategy is defined as follows: the 3D bounding box around the object is discretised into a grid of cell size $4cm \times 3cm$ corresponding to the size of the tactile sensor array. The cells are explored sequentially starting from the edge closest to the robot base. The random strategy follows similar to active strategy, with the next touch chosen randomly among possible touch actions instead of using information gain metric. In order to fairly compare the strategies, we only select the first 10 touches. We set N^T_{min} to 10 points in our experiments. We compare the acquisition strategies from the third touch onwards and report the median (solid line) and median absolute deviation (shaded region) in Figure 5(b). We set the confidence threshold ζ to stop active tactile exploration for recognition at 0.8 or 80%. We report that on average, active approach takes around 14 touches, random approach around 19 touches and uniform approach takes 27 touches to reach a classification accuracy over the confidence threshold.

C. Discussion

As seen from Table III, our proposed network has an average accuracy of 81.25%. Our network has an accuracy over 80% for 9 out of 12 objects. The objects with lower accuracy include (i) object 2 (coffee bottle) at 75%, (ii) object 8 (shampoo) at 20% and (iii) object 10 (sugar box) at 45%. The shampoo is confused with object 1 (cleaner bottle) due to the similar shape and curvature. In fact, if the tactile sensor does not acquire data around the head of the two bottles, due to the sparsity of the tactile data, the model is confused. The sugar box is confused with the tap (object 11). Although the shapes are different, the inaccuracy is due to the fact that the rigidity of the tactile sensor array does not accurately capture high curvatures present in the tape and the sugar box undergoes minor deformations while performing tactile data acquisitions. We notice from Table IV that our approach outperforms the state-of-the-art method [16] by over 50%. The reduced accuracy of [16] is due to the fact that we relax an important assumption in the paper by using dense visual point clouds and sparse tactile point clouds directly without equalising the number of points. In addition, the baseline method [16] was proposed primarily for quasi-planar objects while we use a dataset comprised of 3D objects of varying shape complexity. Furthermore, by leveraging deep neural networks, we are able to extract the discernible features from even sparse point sets by transferring knowledge gained from dense point clouds that hand-crafted features extractors such as CLUE [16] fail to do so. Using our proposed cross-modal transfer learning technique, we note an improvement of accuracy of nearly 30% over the same network without domain adaptation showing the efficacy of our method. Furthermore, our domain adaptation method *VTLoss* combining MMD, CORAL and the classification loss in a weighted linear combination outperforms both MMD and CORAL by over 20%. Similarly, our multi-layer adaptation provides an improved performance of over 10% compared to single-layer adaptation. In fact, we note from the t-distributed stochastic neighbor embedding (t-SNE) [46] visualisations from Figure 4 that the source (visual) features and the target (tactile) features are well clustered after applying domain adaptation. This shows that our model has learnt to effectively discriminate the target features without explicitly training with labelled target data. Our proposed framework is also data efficient. From Table I we note that our active learning approach demonstrates high accuracies within 5% relative error or 2% relative error to that of a fully trained model using just 33% and 58% of the complete dataset respectively. Figure 5(a) shows that our active learning strategy outperforms the random query strategy for visual object learning with fewer data. This demonstrates the amount of labelling efforts saved by adopting the active learning strategy. Similarly, our active tactile object recognition method outperforms the uniform action strategy as seen from Figure 5(b). Using our active tactile approach, the robot can recognise objects with $> 70\%$ accuracy whereas a uniform strategy only reaches 20% accuracy within the first 10 touch actions. The random strategy reaches around 60% accuracy while having larger variability of the recognition as expected from a randomised approach. This helps reducing the overall time for the task execution as robotic tactile action execution is time consuming.

IV. CONCLUSIONS

In this work, we tackle the problem of robotic visuo-tactile cross-modal object recognition leveraging deep neural networks and active perception and learning. Our proposed

[4] https://github.com/viggin/domain-adaptation-toolbox

network *xAVTNet* actively learns from labelled visual point cloud samples and we perform unsupervised cross-modal transfer learning with unlabelled tactile point clouds using our novel domain adaptation *VTLoss* function. Our cross-modal transfer learning method outperforms the state-of-the-art approaches in cross-modal object recognition accuracy. We also demonstrate clear outperformance over baseline strategies with our proposed active learning strategies leading to a reduction in human labelling effort and faster data collection time. Furthermore, our proposed framework uses an active tactile object recognition strategy which leads to data efficiency by reaching high accuracies with fewer data collection steps.

ACKNOWLEDGMENT

We would like to thank Dr. Falco and Ms. Lu for valuable suggestions to reimplement their work for baseline comparisons and Mr. Dutta for helpful discussions. The authors dedicate this work in the memory of Prajval's father, Mr. Murali Rajaram.

REFERENCES

[1] G. Martino *et al.*, "Cross-modal interaction between vision and touch: the role of synesthetic correspondence," *Perception*, 2000.

[2] C. Sann *et al.*, "Perception of object shape and texture in human newborns: evidence from cross-modal transfer tasks," *Developmental science*, vol. 10, no. 3, pp. 399–410, 2007.

[3] T. J. Prescott *et al.*, "Active touch sensing," *Phil. Trans. of the Roy. Soc.*, vol. 366, no. 1581, pp. 2989–2995, 2011.

[4] M. Kaboli, K. Yao, and G. Cheng, "Tactile-based manipulation of deformable objects with dynamic center of mass," in *International Conference on Humanoid Robots (Humanoids)*. IEEE, 2016.

[5] K. Yao, M. Kaboli, and G. Cheng, "Tactile-based object center of mass exploration and discrimination," in *2017 IEEE-RAS 17th International Conference on Humanoid Robotics (Humanoids)*. IEEE, 2017, pp. 876–881.

[6] M. Kaboli, A. De La Rosa T, R. Walker, and G. Cheng, "In-hand object recognition via texture properties with robotic hands, artificial skin, and novel tactile descriptors," 2015, pp. 1155–1160.

[7] F. Liu, S. Deswal, A. Christou, Y. Sandamirskaya, M. Kaboli, and R. Dahiya, "Neuro-inspired electronic skin for robots," *Science Robotics*, vol. 7, no. 67, p. eabl7344, 2022.

[8] Q. Li *et al.*, "A review of tactile information: Perception and action through touch," *IEEE Trans. on Rob.*, vol. 36, no. 6, pp. 1619–1634, 2020.

[9] M. Kaboli, W. Rich, and G. Cheng, "Re-using prior tactile experience by robotic hands to discriminate in-hand objects via texture properties," in *International Conference on Humanoid Robots (Humanoids)*. IEEE, 2016, pp. 2242–2247.

[10] M. Kaboli *et al.*, "A tactile-based framework for active object learning and discrimination using multimodal robotic skin," *IEEE Rob. and Auto. Let.*, 2017.

[11] M. Kaboli *et al.*, "Active tactile transfer learning for object discrimination in an unstructured environment using multimodal robotic skin," *Int. Jour. of Hum. Rob.*, 2018.

[12] M. Kaboli *et al.*, "Tactile-based active object discrimination and target object search in an unknown workspace," *Auto. Rob.*, 2019.

[13] M. Kaboli and G. Cheng, "Robust tactile descriptors for discriminating objects from textural properties via artificial robotic skin," *IEEE Trans. on Rob.*, vol. 34, no. 4, pp. 985–1003, 2018.

[14] R. Dahiya *et al.*, "Large-area soft e-skin: The challenges beyond sensor designs," *Proc. of the IEEE*, vol. 107, no. 10, pp. 2016–2033, 2019.

[15] F. Liu *et al.*, "Printed synaptic transistor–based electronic skin for robots to feel and learn," *Science Robotics*, vol. 7, no. 67, p. eabl7286, 2022.

[16] P. Falco *et al.*, "A transfer learning approach to cross-modal object recognition: from visual observation to robotic haptic exploration," *IEEE Tran. on Rob.*, 2019.

[17] B. Gong *et al.*, "Geodesic flow kernel for unsupervised domain adaptation," in *Conf. on comp. vis. and pat. recog.* IEEE, 2012.

[18] B. S. Zapata-Impata *et al.*, "Generation of tactile data from 3d vision and target robotic grasps," *IEEE Trans. on Hap.*, vol. 14, no. 1, pp. 57–67, 2020.

[19] Y. Li *et al.*, "Connecting touch and vision via cross-modal prediction," in *Conf. on Comp. Vis. and Pat. Recog.*, 2019, pp. 10 609–10 618.

[20] D. Feng *et al.*, "Active prior tactile knowledge transfer for learning tactual properties of new objects," *Sensors*, vol. 18, no. 2, p. 634, 2018.

[21] Y. Guo *et al.*, "Deep learning for 3d point clouds: A survey," *IEEE Trans. on Pat. Ana. and Mac. Int.*, 2020.

[22] C. R. Qi *et al.*, "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *Conf. on Comp. Vis. and Pat. Recog.*, 2017, pp. 652–660.

[23] C. R. Qi *et al.*, "Pointnet++: Deep hierarchical feature learning on point sets in a metric space," *Adv. in Neu. Inf. Proc. Sys.*, vol. 30, 2017.

[24] C. Xiao and J. Wachs, "Triangle-net: Towards robustness in point cloud learning," in *Win. Conf. on App. of Comp. Vis.*, 2021, pp. 826–835.

[25] P. K. Murali *et al.*, "Intelligent in-vehicle interaction technologies," *Adv. Int. Sys.*, vol. 4, no. 2, p. 2100122, 2022.

[26] P. K. Murali *et al.*, "Towards robust 3d object recognition with dense-to-sparse deep domain adaptation," in *Int. Conf. on Flex. and Print. Sens. Sys.* IEEE, 2022, pp. 1–4.

[27] P. K. Murali *et al.*, "An empirical evaluation of various information gain criteria for active tactile action selection for pose estimation," in *Int. Conf. on Flex. and Print. Sens. Sys.* IEEE, 2022, pp. 1–4.

[28] G. Wilson and D. J. Cook, "A survey of unsupervised deep domain adaptation," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 11, no. 5, pp. 1–46, 2020.

[29] M. Wang and W. Deng, "Deep visual domain adaptation: A survey," *Neurocomputing*, vol. 312, pp. 135–153, 2018.

[30] K. M. Borgwardt *et al.*, "Integrating structured biological data by kernel maximum mean discrepancy," *Bioinformatics*, 2006.

[31] B. Sun *et al.*, "Return of frustratingly easy domain adaptation," in *Proc. of AAAI Conf. on Art. Int.*, vol. 30, no. 1, 2016.

[32] P. K. Murali *et al.*, "Active visuo-tactile point cloud registration for accurate pose estimation of objects in an unknown workspace," in *Int. Conf. on Int. Robots and Sys.* IEEE, 2021.

[33] M. M. Zhang *et al.*, "Active end-effector pose selection for tactile object recognition through monte carlo tree search," in *Int. Conf. on Int. Robots and Sys. (IROS)*. IEEE, 2017.

[34] P. K. Murali *et al.*, "Active visuo-tactile interactive robotic perception for accurate object pose estimation in dense clutter," *IEEE Rob. and Auto. Letters*, 2022.

[35] D. Feng *et al.*, "Deep active learning for efficient training of a lidar 3d object detector," in *Int. Veh. Symp. (IV)*. IEEE, 2019.

[36] W. H. Beluch *et al.*, "The power of ensembles for active learning in image classification," in *IEEE conf. on comp. vis. and pat. recog.*, 2018, pp. 9368–9377.

[37] Y. Gal *et al.*, "Dropout as a bayesian approximation: Representing model uncertainty in deep learning," in *Int. Conf. on Mac. Learn.* PMLR, 2016, pp. 1050–1059.

[38] A. Gretton *et al.*, "A kernel two-sample test," *The Jour. of Mac. Learn. Res.*, vol. 13, no. 1, pp. 723–773, 2012.

[39] A. Gretton *et al.*, "Optimal kernel choice for large-scale two-sample tests," in *Adv. in Neu. Inf. Proc. Sys.* Citeseer, 2012, pp. 1205–1213.

[40] B. Sun and K. Saenko, "Deep coral: Correlation alignment for deep domain adaptation," in *European conference on computer vision*. Springer, 2016, pp. 443–450.

[41] J. Yosinski *et al.*, "How transferable are features in deep neural networks?" *Adv. in Neu. Inf. Proc. Sys.*, vol. 27, pp. 3320–3328, 2014.

[42] F. Bourgault *et al.*, "Information based adaptive robotic exploration," in *Int. Conf. on Int. Rob. and Sys.* IEEE, 2002.

[43] A. Hornung *et al.*, "Octomap: An efficient probabilistic 3d mapping framework based on octrees," *Autonomous robots*, 2013.

[44] C. Potthast *et al.*, "A probabilistic framework for next best view estimation in a cluttered environment," *Jour. of Vis. Comm. and Img. Rep.*, vol. 25, no. 1, pp. 148–164, 2014.

[45] P. K. Murali *et al.*, "In situ translational hand-eye calibration of laser profile sensors using arbitrary objects," in *Int. Conf. on Rob. and Auto.* IEEE, 2021, pp. 11 067–11 073.

[46] L. Van Der Maaten, "Accelerating t-sne using tree-based algorithms," *The Jour. of Mac. Learn. Res.*, vol. 15, no. 1, pp. 3221–3245, 2014.