IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 5 February 2025, accepted 5 April 2025, date of publication 29 April 2025, date of current version 16 May 2025.

Digital Object Identifier 10.1109/ACCESS.2025.3565330

RESEARCH ARTICLE

Training-Free Affordance Labeling and Exploration Using Subspace Projection and Manifold Curvature Over Pre-Trained Deep Networks

İSMAİL ÖZÇİL[1] **AND A. BUĞRA KOKU**[1,2]
[1]Department of Mechanical Engineering, Middle East Technical University (METU), 06800 Ankara, Türkiye
[2]Center for Robotics and Artificial Intelligence (ROMER), METU, 06800 Ankara, Türkiye
Corresponding author: İsmail Özçil (iozcil@metu.edu.tr)

ABSTRACT The advancement in computing power has significantly reduced the training times for deep learning, enabling the rapid development of networks designed for object recognition. However, the exploration of object utility, the object's affordance, as opposed to object recognition, has received comparatively less attention. Existing object affordance models exhibit shortcomings, including limited robustness across diverse architectures and insufficient performance in complex environments. This work focuses on using pre-trained networks trained on object classification datasets to explore object affordances. While these networks have proven instrumental in transfer learning for classification tasks, the presented approach in this study diverges from conventional object classification methods by labeling affordances without modifying the final layers. Instead, pre-trained networks are employed to learn affordance labels without requiring specialized classification layers. Two approaches are tested: the Subspace Projection Method and the Manifold Curvature Method, which facilitate the determination of affordance labels without such modifications. Both the Subspace Projection Method and the Manifold Curvature Method were evaluated using nine distinct pre-trained networks across two different affordance datasets. The Subspace Projection Method achieved a True Positive Rate of up to 94% and 96% for the best-performing networks on each dataset, while the Manifold Curvature Method attained True Positive Rates exceeding 98% and 99% with its top-performing networks. Furthermore, both methods identify affordance labels that are not marked in the ground truth but are present in various cases. The robustness of the Manifold Curvature Method and the exploration capability of both methods highlight the effectiveness of proposed techniques for affordance labeling.

INDEX TERMS Affordance, deep learning, manifold curvature, subspace clustering.

I. INTRODUCTION

Object recognition involves identifying objects using sensor data, typically captured through images or videos from a camera. For a robot to interact effectively with its environment, it needs to recognize and detect objects in its sensor space and then use these recognized and detected objects in its parameter space when required.

The concept of "affordance", introduced by Gibson [1], is essential in this context. Affordance refers to a specific combination of an object's substance and the surfaces about an animal. Later, Gibson modified this concept to relate the perception of an object to its potential actions [2]. Affordance detection is identifying how an object can be used based on its physical properties. Affordance detection plays a pivotal role in autonomous robotics and human-robot interaction and offers a more dynamic understanding of object functionality. However, detecting multiple, potentially

The associate editor coordinating the review of this manuscript and approving it for publication was Massimo Cafaro.

overlapping affordances is a complex task. This study proposes two innovative methodologies, Subspace Projection Method (SPM) and Manifold Curvature Method (MCM), that use pre-trained networks for affordance labeling without transfer learning, offering a new perspective on affordance detection.

A. OBJECTIVE

The primary objective of this study is to develop methodologies that allow for affordance detection using pre-trained networks without the need for retraining or adding new layers to pre-trained networks. Conventional approaches often rely on transfer learning, where new layers are added and trained to suit the networks to affordance tasks. This study, however, focuses on extracting and utilizing feature vectors directly from pre-trained networks like ResNet-18, ResNet-50 [3], ResNext-101 [4], and others, bypassing any additional training requirements. By using SPM and MCM, the presented approach in this study enables the labeling of affordances, including those not explicitly provided in ground truth information of the datasets, hence enhancing the exploration of hidden affordances. The novelty of the methods presented in this study lies in their flexibility and scalability, applicability across diverse architectures, and lack of modifications to the networks. One-hot encoding, commonly used in classification tasks to map each class to a unique binary vector, is not sufficient for affordance detection. Objects often afford multiple actions, leading to overlapping affordance labels. The proposed methods account for this complexity by utilizing feature vectors rather than binary classification schemes, allowing for more suitable labeling decisions.

B. RELATED WORK

Affordance learning has been explored through various approaches, from traditional robotics to advanced machine learning models. For instance, Uğur et al. [5] developed a framework that enables robots to mimic human infant motor skills. They focused on affordance learning through sensorimotor interactions. Their approach relied on behavioral primitives. Similarly, Modayil and Kuipers [6] and Koppula et al. [7] presented unsupervised methods for affordance learning using sensorimotor data and Markov random fields using RGBD videos. Similarly, Sawatzky et al. [8] introduced a convolutional neural network (CNN) for multilabel affordance segmentation using keypoint annotations. While their contributions are significant, they depend on transfer learning, which requires modifying the last layers and training the resulting modified network layers. The proposed methods presented in this study overcome this limitation by directly utilizing pre-trained networks. Thermos et al. [9] obtained pixel-level affordances of images by training an auto-encoder network with human-object interaction videos. Nguyen et al. [10] employed CNNs to detect object affordances. Similarly, these studies also depend on training a network. In the context of affordance learning for robotic manipulation tasks, Iriondo et al. [11] introduced a technique to detect grasping points using graph convolutional networks and point cloud data. Wu and Chirikjian [12] explored strategies for predicting pourability affordances using RGBD cameras. Myers et al. [13] utilized RGBD images to extract feature vectors and made affordance decisions through Support Vector Machine and Structured Random Forest methods. Thermos et al. [14] developed a model to detect the affordances of object parts in RGBD videos. However, their reliance on pixel-level affordance segmentation and depth data limits the generalizability of their models. The presented approach in this study diverges by focusing solely on RGB data for feature extraction, bypassing the need for additional data inputs such as depth images. Moreover, the proposed methods do not require retraining when introducing new affordance groups, which allows for a more flexible and scalable approach to affordance labeling. For affordance detection from visual data, Li et al. [15] proposed LOCATE, a framework that focuses on localizing and transferring object parts for weakly supervised affordance grounding. Meanwhile, Vo et al. [16] address the open vocabulary challenge by proposing a method for affordance detection using knowledge distillation and text-point correlation, enabling affordance learning even with limited labeled data. These methods, while innovative, still use transfer learning techniques that require retraining layers of pre-trained networks to adapt them to affordance-specific tasks. Mur-Labadia et al. [17] focused on affordance mapping from egocentric vision, demonstrating the applicability of multi-label classification techniques in affordance detection. Some approaches have focused on predicting 3D affordances from 2D images, and proposed methods to ground 3D object affordances from 2D human-object interaction images. Yang et al. [18] present an approach for learning 3D human-object interaction relations from 2D images. Kim et al. [19] explored using pre-trained 2D diffusion models for 3D affordance discovery. Their findings showed promise in transferring knowledge from 2D to 3D affordance tasks without extensive training. Ragusa et al. [20] focus on developing efficient models for wearable robots, proposing tiny networks for affordance segmentation on resource-constrained devices. Ardón et al. [21] focused on robot-to-human object handovers, considering human comfort and object usability.

There are also studies about affordances for autonomous driving. Chen and Chenyi [22] proposed a method to map an input image to driving affordances using ConvNet, providing a more controllable approach to autonomous driving compared to the direct mapping of commands. Ransikarbum et al. [23] developed a model for driver decision-making using road affordances to enhance autonomous driving systems.

The concept of affordance is also investigated by several studies. Bozeat et al. [24] studied the effect of prior

knowledge on object usage by patients and found that prior knowledge has a significant impact on object usage. Federico and Brandimonte [25] investigated tool usage characteristics of people and found support for reasoning-based theories of human tool use. Şahin et al. [26] discussed the affordance concept and proposed a new formalization for autonomous robots.

On the design side, in addition to the studies that aim to obtain affordance information of objects, Andries et al. [27] designed 3D objects based on required functionalities by training a neural network to relate function to form using a dataset of affordance-labeled objects.

Apicella et al. [28] focused on reproducibility challenges in affordance segmentation. Their incorporation of methods like Mask2Former showcased improvements in accuracy; however, sensitivity to scale variations remained a limitation.

Various approaches have been proposed for learning affordances from visual data. Chen et al. [29] present a method for 6-DoF grasp detection that uses both implicit neural representation and visual affordance estimation. Li et al. [30] introduced a framework capable of generalizing affordance detection to novel objects with one-shot learning. The study highlighted the utility of pre-trained models in extracting semantically rich features, although their approach focuses primarily on segmentation-based affordances. Ruiz and Mayol-Cuevas [31] used interaction tensors to estimate affordance possibilities for various actions, but predicting affordances for flexible objects was challenging. Using statistical relational learning, Moldovan et al. [32] learned affordances for multiple interacting objects. Uğur et al. [33], [34], [35] utilized robotic hand actions and observations to learn object affordances, incorporating behavioral parameters and recognizing traversable objects in a room. Xu et al. [36] introduced a method for predicting future affordance states following an action applied to an environment. Hassanin et al. [37] made improvements to Mask-RCNN to address scaling issues in determining object part affordances at various scales. Pandey and Alami [38] proposed a framework to enhance human-robot interaction by using affordances to perform contextually appropriate and relevant actions. Ragusa et al. [39] presented a method for real-time affordance detection on resource-constrained systems. There are also recent works focused on affordances. Li et al. [40] introduced LASO, a language-guided affordance segmentation task, focused on the need for a semantic understanding of affordances in 3D data. They combined large language models (LLMs) with 3D data for semantic affordance understanding and showed that incorporating semantic cues can enhance affordance learning outcomes. While LASO advanced semantic guidance, it primarily focused on 3D affordance segmentation, contrasting with our image-level labeling methodology. Similarly, Luo et al. [41] focused on interactive affinities in affordance learning, extracting cues from human-object interactions to improve model performance in diverse environments. Xu et al. [42] introduced a vision-language model designed to enhance robotic manipulation tasks. Their integration of fine-grained language cues improved semantic affordance recognition, especially in complex scenarios.

Regarding datasets and resources for affordance learning, Khalifa and Shah [43] introduced a large-scale multi-view RGBD dataset containing object images with affordance labels. Their dataset offers a comprehensive range of affordance labels, but their reliance on pixel-level segmentation techniques adds computational overhead. Chuang et al. [44] constructed a dataset with people interacting with objects and offered a new method to determine object affordances and human-object interactions. Mur-Labadia et al. [17] proposed a novel multi-label affordance mapping technique that leverages egocentric vision to achieve pixel-level segmentation. Their work resulted in the EPIC-Aff dataset, which is significant for benchmarking affordance detection models.

In the broader context of multi-label classification, Wang et al. [45] combined convolutional and recurrent neural networks to jointly model image features and label dependencies. Their method achieved state-of-the-art mean average precision (mAP) scores of 84.0% on PASCAL VOC 2007 and 61.2% on MS COCO, showcasing the importance of capturing label correlations in multi-label classification tasks. The framework's success in exploiting label dependencies aligns with our objective of handling overlapping affordances in image-level labeling. Later, Li et al. [46] proposed a smooth pairwise ranking loss and a label decision module to enhance multi-label classification. On PASCAL VOC 2007, it achieved an mAP of 90.3%. This aligns with our focus on improving affordance recognition by leveraging dependencies between labels in multi-affordance scenarios. Schultheis et al. [47] provided a statistical framework for multi-label classification metrics requiring exactly k labels per instance. Their approach, utilizing a Frank-Wolfe-based learning algorithm, emphasized balanced utility across labels and addressed challenges in extreme classification scenarios with long-tail label distributions. This study contrasts with our work by focusing on label constraints, whereas our methodology emphasizes scalable, segmentation-free affordance detection. Prokofiev and Sovrasov [48] introduced a method integrating metric learning with attention mechanisms. Their approach achieved a mAP of 96.70% on the PASCAL VOC 2007 dataset using TResNet-L, highlighting its efficiency and accuracy. This method highlights label dependency modeling, aligning with the need for robust handling of multi-affordance labeling tasks in complex environments.

Xu et al. [49] introduced the ADDS framework and achieved a state-of-the-art result for multi-label classification. It utilizes a dual-modal decoder to align visual and textual features effectively. A Pyramid-Forwarding technique achieves high performance on large-scale datasets like MS COCO, attaining an mAP of 93.54% using the ViT-L-336 model at a resolution of 1344×1344 for input images.

Liu et al. [50] introduced a framework that uses transformer decoders to query the existence of class labels. This method utilizes learnable label embeddings as queries to extract class-related features via cross-attention mechanisms, enabling adaptive and discriminative feature extraction for each label. Their approach demonstrated superior performance across multiple datasets, achieving a mAP of 91.3% on MS-COCO and a state-of-the-art mAP 97.3% on PASCAL VOC 2007. Zhu et al. [51] introduced a novel framework that incorporates scene awareness into label graph learning. This approach enhances the relationships between labels by leveraging the scene context of the image, achieving state-of-the-art results on benchmarks such as COCO and Pascal VOC. Their work highlights the value of integrating scene-level information for improving multi-label classification accuracy. Jia et al. [52] proposed a framework to learn disentangled representations for each label, addressing the challenges of shared feature representations in multi-label classification. This method ensures improved label-specific feature extraction, achieving competitive results across several datasets.

Koku et. al. [53] investigated the usage of the pre-trained deep neural networks as feature extractors without any further training. They suggested deep CNNs trained on large and diverse datasets can be treated as universal feature extractors. They showed that such networks when used as feature extractors, can effectively perform clustering of unknown image categories using unsupervised methods. Sekmen et al. [54] proposed methods for enhancing the separability of feature spaces in deep networks. By introducing adaptive projection matrices and emphasizing robust feature subspace separation, their approach improves classification performance. These methods provide insights for improving feature representation in multi-affordance scenarios where distinct labeling is critical. Aldroubi et al. [55] presented a framework for CUR matrix decomposition to improve data representation and clustering efficiency. Their insights into leveraging low-rank approximations and similarity matrices for subspace clustering provide a theoretical foundation relevant to our manifold curvature method, which aims to enhance subspace separation for robust affordance labeling.

In summary, while the existing literature has made notable strides in affordance learning, most approaches rely on modifications to pre-trained networks i.e. transfer learning, pixel-level segmentation, or specific datasets. This study differentiates itself by employing pre-trained networks without transfer learning, using SPM and MCM to explore affordances. This avoids the overhead of retraining and segmentation while providing flexibility in affordance detection across various object classes.

C. CONTRIBUTIONS OF THE STUDY

Given that transfer learning-based affordance labeling methods start with a pre-trained deep network (generally a network trained on ImageNet [56] dataset), after adapting the last layer to the labels in the dataset training is performed. Evidently, this training changes the weights of the pre-trained network, and as transfer learning progresses, the network adopts the affordance dataset data. Hence, even if feature vectors are extracted from this dataset corresponding to an image from the affordance dataset, these vectors will be different from the vectors that would be obtained if the same images were fed to the unaltered pre-trained network.

The originality of this work relies on the fact that labeling methods presented in this study do not require any training at all. Feature vectors directly obtained from a pre-trained are used. Although the pre-trained network has not seen any affordance labels at all, this work shows that extracted vectors can be clustered into a completely new set of labels (i.e. affordance labels).

It should also be noted that the ImageNet [56] dataset poses a one-hot-encoded labeling problem. However, affordance labeling involves one or many labels associated with an image. Proposed training methods tackle the problem of multi-labeling (many-hot-encoding) problem using a network trained on a one-hot-encoded dataset. It is shown that the proposed training-free MCM outperforms the state-of-the-art solutions that use further training.

The contributions of this work can be listed as follows:

- Applicability of Pre-Trained Networks Without Transfer Learning: This study demonstrates pre-trained networks like ResNet [3], RegNetY [57], and EfficientNet [58] trained on a large dataset like ImageNet [56] can be used for affordance detection without the need for transfer learning. This approach avoids the computational expense and time associated with training models from the ground up or modifying and retraining existing models, offering a more efficient approach.
- Introduction of SPM and MCM: Two methods for affordance labeling are introduced and explained. These two methods use feature vectors extracted from pre-trained networks for affordance labeling. SPM groups feature vectors corresponding to the same affordances, while MCM calculates the local angle of feature space manifolds to detect potential affordances. Both methods offer flexibility in labeling objects with multiple affordances and exploring hidden affordances not explicitly labeled in datasets.
- Exploration of Hidden Affordances: The methods proposed in this study enable the detection of affordances that are not marked in the ground truth of datasets. This capability allows for a deeper exploration of an object's functional potential, offering significant benefits in real-world robotics applications where the affordances of an object are not obvious.
- Note that this approach uses a pre-trained network to classify images that have affordance labels. Given that, the proposed methods (especially MCM) use a pre-trained network directly and do not update or alter it to adapt to the affordance labeling problem, this method also has the potential to use existing networks trained on

ImageNet in other multi-labeling or many-hot-encoding problems without resorting to transfer learning.

These contributions collectively enhance the understanding of affordance detection and exploration. An indirect contribution of this work is on the problem of multi-labeled classification problems, where there is a many-to-many mapping between inputs and the labels.

The paper is structured as follows: Section II presents the dataset used in the study and details the proposed methodologies, including the mathematical foundations of SPM and MCM. Section III analyzes the results, comparing the performance of SPM and MCM across different pre-trained architectures. Finally, Section IV discusses potential real-world applications of the proposed methods and suggestions for future research.

II. METHOD

This study uses two different affordance datasets containing RGBD images of various objects, accompanied by segmentation labels that specify affordance labels for each pixel. While most related works discussed in the preceding section relied on depth images to estimate object affordances, the presented methods in this study take a different path by omitting depth image data. The depth data is omitted to simplify the affordance labeling process. By focusing on RGB images alone, the study ensures that the methods are more widely applicable, as it allows the use of standard RGB cameras instead of more complex and expensive depth-sensing cameras. This simplification increases the potential use cases of the proposed methods, enabling them to be applied in real-world scenarios where depth sensors may not be available or practical.

There exist two types of affordance labeling methods. The first method requires segmenting and labeling individual pixels within an image based on their affordances. The second method, which is the focus of this study, does not make pixel or segment labeling and instead assigns predicted affordances directly to the entire image. To validate the proposed methodologies, image-level labeling without any segmentation is focused. By focusing on image-level labeling, the study aims to provide a scalable and flexible solution that can be easily implemented across various datasets and scenarios, making it a more generalizable approach to affordance detection.

In addition to the affordance dataset they provided, Khalifa and Shah [43] presented various approaches, some of which involved employing pre-trained networks for feature extraction. Typically, these methods entailed appending and training a new fully connected layer to generate estimations tailored to the specific dataset. In contrast, the proposed methods differ from this practice by only using pre-trained networks to obtain feature vectors. Unlike methodologies that incorporate and train new fully connected layers, this study avoids directly obtaining results from the tailored fully connected layer. Instead, the feature vector derived from the last layer before the fully connected layer of the original network is harnessed to formulate estimation criteria through two distinct methods. The overall process of affordance labeling proposed in this study is shown in Figure 1. The rationale behind selecting feature vectors lies in their capacity to afford flexibility in the number of affordance decisions, allowing for easier integration of new affordance classes into an existing decision structure. Furthermore, utilizing vectors facilitates the exploration of novel affordances associated with a given object, predicated on the "closeness" to other object vectors with different affordance labels. Additionally, adopting vectors for decision-making allows us to explore manifolds where feature vectors of various affordance classes are situated.

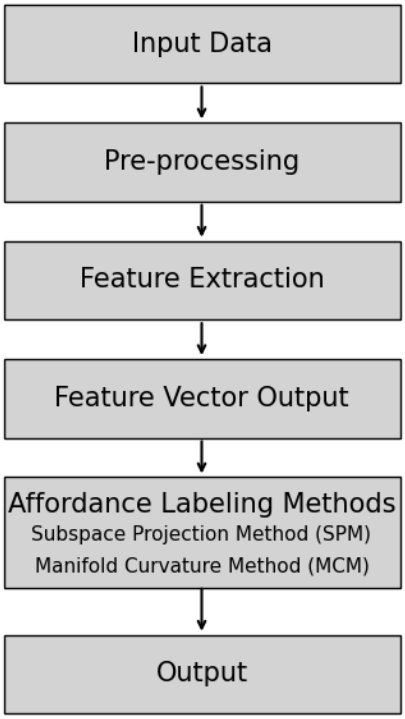

FIGURE 1. Affordance labeling process.

A. A LARGE SCALE MULTI-VIEW RGBD VISUAL AFFORDANCE LEARNING DATASET

The dataset named 'A large scale multi-view RGBD visual affordance learning dataset [43], which provides object affordance annotations at the pixel level to test the proposed methods is employed in this study. The dataset is composed of 23605 scenes, with every scene having one RGB, one depth, and one label image from 37 object categories annotated with 15 affordance categories, which are 'grasp', 'wrap grasp', 'containment', 'liquid-containment', 'openable', 'dry', 'tip-push', 'display', 'illumination', 'cut', 'pourable', 'rollable', 'absorb', 'grip' and 'stapling'.

The 'grasp' affordance refers to parts of objects that can be grasped by the user, while 'wrap grasp' refers to objects that can be held by wrapping fingers around them. 'Containment' is associated with bowl-like objects capable of containing other materials, and 'liquid-containment' denotes bottle-like parts of objects that can hold liquids. For consistency, 'contain' and 'liquid contain' are used instead of 'containment' and 'liquid-containment', respectively. The 'openable' affordance refers to cap-like or lid-like parts of objects that the user can open. For consistency, 'open' is used instead of 'openable' throughout this study. The 'dry' affordance identifies parts of objects used to dry other surfaces or objects. 'Tip-push' describes button-like parts of objects that can be pressed, while 'display' identifies objects equipped with displays. 'Illumination' refers to objects that emit light. For consistency, 'illuminate' is used instead of 'illumination'. The 'cut' affordance applies to objects capable of cutting other objects, and 'pourable' describes objects designed to pour liquids. 'Pour' instead of 'pourable' for consistency. 'Rollable' refers to objects that can be rolled. 'Roll' is used instead of 'rollable'. The 'absorb' affordance identifies objects with porous structures that can absorb liquids, and the 'grip' pertains to objects with parts designed for a secure grip. Finally, 'stapling' refers to staples. For consistency, 'staple' is used instead of 'stapling'.

The dataset comprises scene images in which a single object is presented. These objects are pixel-level annotated with corresponding affordance labels in their label images. Parts of the objects are labeled with suitable affordance labels. Due to the pixel-level annotation, each region can be labeled with only one affordance label. However, it is important to note that an object's individual part or region may afford multiple actions. This opens up the possibility to ''explore'' additional affordances, especially those not explicitly labeled, but present within the dataset. For example, an object in the image may have side surfaces that are both suitable for a 'wrap-grasp' and a 'rollable' action. Due to the pixel-level annotation, only one affordance label can be assigned to each region (in this case, one of these affordances is labeled for the side surface pixels). Nevertheless, this situation highlights the opportunity to investigate additional affordances, including those that might not be immediately obvious from the dataset's ground truth.

In addition to the scenes with single objects, the dataset also contains 35 cluttered/complex test scenes with different objects and multiple affordances. Figure 2 illustrates some examples from the dataset. Although not used in any part of the study, depth images are scaled to be represented by a grayscale image to be shown as a dataset example. Notably, the image dimensions across the dataset vary; different scenes may exhibit distinct width and height values. Occasionally, some images possess dimensions incompatible with pre-trained networks. Some images may not be compatible with the fixed input requirements of pre-trained networks. To address this issue, images with different dimensions are padded to become rectangular. Zero padding is applied to both sides of the image, ensuring that the object remains centered. Once the images are adjusted to have uniform dimensions, they are rescaled as part of the preprocessing pipeline. As a result, before sending them as input to a pre-trained network, RGB images are first normalized, and then, if the image dimensions are not square, padding is applied to make the width and height equal, preserving the central alignment of the object. The final processed images are standardized to a resolution of 224×224 pixels, making them suitable for input into the pre-trained networks. This preprocessing step ensures that all images conform to the input requirements of the models, facilitating consistent and efficient feature extraction.

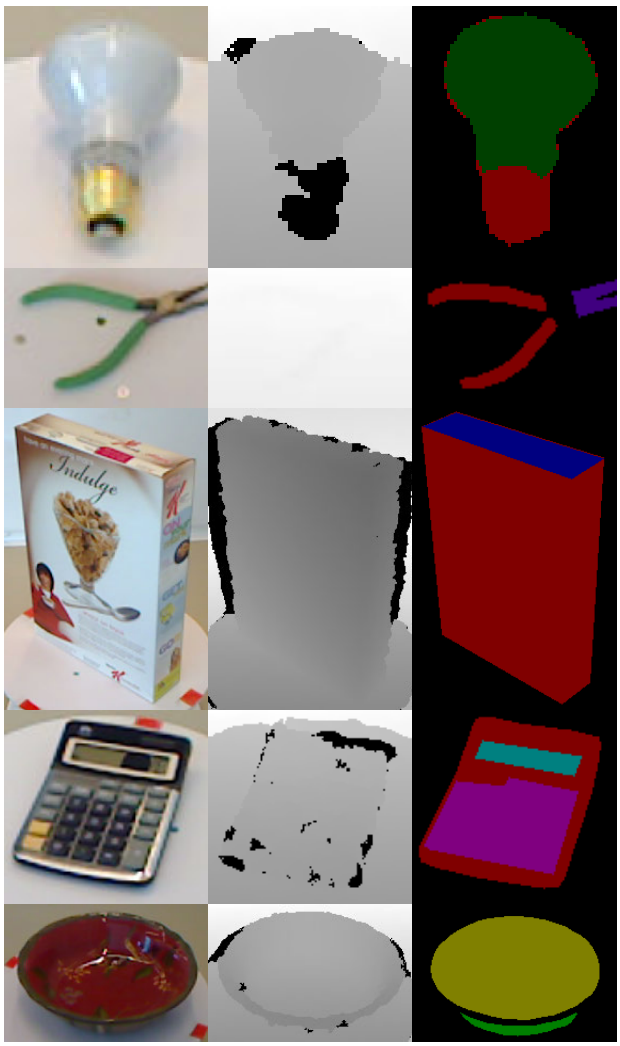

FIGURE 2. A large scale multi-view RGBD visual affordance learning dataset examples of the objects with RGB, depth, and label images [43].

B. RGB-D PART AFFORDANCE DATASET

To further evaluate the performance of the proposed methods, RGB-D Part Affordance Dataset [13] is employed. This dataset is specifically designed to provide RGBD images annotated with affordance labels. The dataset consists of

7 affordance categories: 'grasp', 'cut', 'scoop', 'contain', 'pound', 'support', and 'wrap-grasp' each associated with distinct object parts that afford these actions. Figure 3 shows the RGB-D Part Affordance dataset examples. Similar to Figure 2, depth images are scaled to be shown by a grayscale image. Since labels are given by an array file, labels are converted to an RGB file. Unlike the A large scale multi-view RGBD visual affordance learning dataset [43], image sizes are the same across the dataset. However, similarly, image preparation is done by scaling and normalizing before the feature extraction process. The dataset provides a comprehensive set of annotations, with each image containing an RGB image, a depth image, and a corresponding affordance label map. In addition to the pixel-wise segmented label map, an ordered map for each pixel is also provided. The affordances are labeled at the part level, meaning each object part is associated with a specific affordance. For example, grasp affordance refers to parts of objects that can be easily gripped, while support denotes parts that provide support or stability to the object.

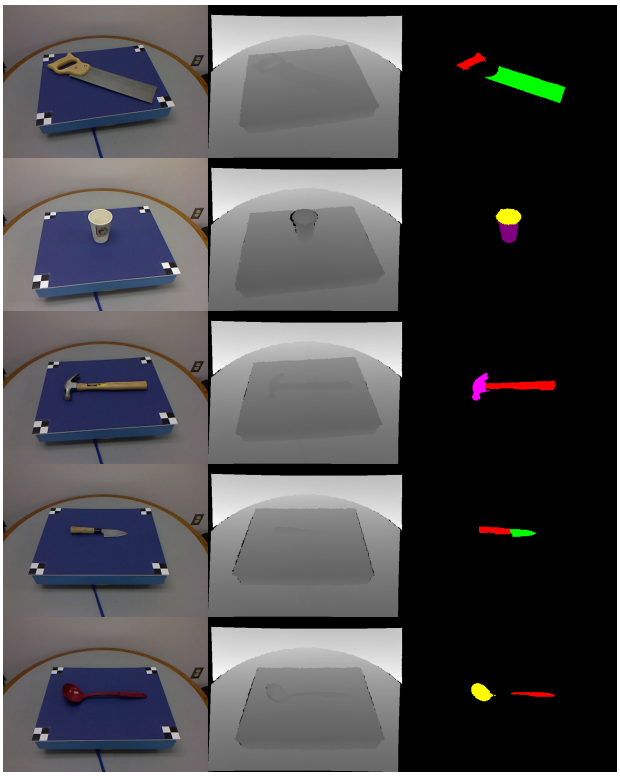

FIGURE 3. RGB-D Part Affordance Dataset examples of the objects with RGB, depth, and label images.

The RGB-D Part Affordance Dataset includes a total of 28843 RGBD images, each containing one object. Moreover, some cluttered scene examples are also provided in the dataset.

By integrating this dataset, the generalization capabilities of SPM and MCM are aimed to be tested. These methods will be tested across the 7 affordance categories, allowing us to evaluate their ability to accurately predict and label the affordances associated with different object parts. Moreover, comparing the performance of these methods on two different affordance datasets is also important to understand the strengths and weaknesses of the proposed methods.

C. PRE-TRAINED NEURAL NETWORK

As pre-trained networks, ResNet-18, ResNet-50, ResNet-101, ResNet-152 [3], ResNext-101 [4], RegNetY [57], Efficient-NetV2 [58], ViT-L/16, and ViT-B/16 [59] trained on the ImageNeT [56] dataset have been selected to generate feature vectors from the previously explained affordance dataset. ImageNet [56] is a dataset used for object recognition tasks. The final fully connected layers of these pre-trained networks are removed to obtain feature vectors as outputs, as shown in Figure 4. Due to the distinctive architectures of these networks, the dimensions of their respective feature vectors vary. As explained in the preceding section, the dataset contains RGB images paired with depth images and corresponding affordance annotations at the pixel level. In this context, only the RGB images of objects serve as inputs for the modified networks to extract feature vectors. The depth images are disregarded throughout this study and are not used at any stage. While pixel-level object usability labels are not directly utilized, they serve as image-level labels for the experiments of this study. Each object image is annotated with a combination of object usability labels derived from its pixel annotations rather than using the pixel-level annotations themselves. Multiple networks are employed to evaluate the labeling performance of feature vector outputs generated by each network. The utilization of various networks enables the evaluation of labeling performances and the robustness of labeling methods on different pre-trained network architectures. Developing a technique that performs well with all the previously mentioned networks is aimed. This approach enables a comprehensive analysis of the effectiveness and stability of the labeling techniques under consideration.

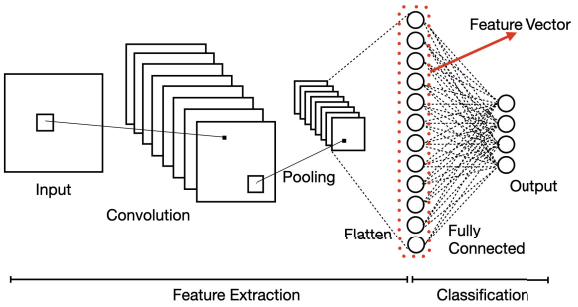

FIGURE 4. A simple CNN with feature vector extraction.

A dataset is generally divided into two parts for deep learning applications: training set and validation or test set. Since the pre-trained neural network is planned to be used as

it is without its last fully connected layer, there is no need for training for this case. Hence, the dataset is divided into learning and validation parts. Decisions are made according to the properties of the data extracted from a learning part of the dataset. ''Learning'' is used here since there is no layer-by-layer training in the presented methods, where learning involves the analysis of feature vectors obtained from the part of the dataset set apart for this purpose. Of 23,605 scenarios from the A large scale multi-view RGBD visual affordance learning dataset, 10,000 images are allocated for learning, while 13,605 images are reserved for validation. Similarly, out of 28,000 scenarios of RGB-D Part Affordance dataset, 10,000 images are allocated for learning, and the remaining are reserved for validation. For both datasets, less than half of the dataset is used for the learning. The feature vectors corresponding to the learning data are then analyzed and categorized based on the affordance labels associated with each dataset entry. Algorithm 1 shows the process of generating affordance groups from the affordance labels using the learning set. It is important to acknowledge that a single feature vector may be linked to multiple affordance categories, reflecting the potential for an object to serve various purposes and, consequently, carry multiple labels.

Algorithm 1 Dataset Feature Extraction

Input: r: RGB images,
$label\_img$: Affordance segmented label images,
$affordancess$: List of all affordance labels
Output: $C_{learning}$, $C_{validation}$:List of learning and validation feature vector sets.
Require: f : Pre-Trained Network as Feature Extractor.

```
Separate r to r_learning and r_validation
Initialize C_learning
for i in affordances do
    Initialize C_{i,learning}.
    for scene_image in r_learning do
        if label_img(scene_image) has i then
            Append f(scene_image) to C_{i,learning}
        end if
    end for
    Append C_{i,learning} to C_learning
end for
Initialize C_validation
for scene_image in r_validation do
    j ← f(scene_image)
    Append j to C_validation
end for
return C_learning, C_validation
```

D. SUBSPACE PROJECTION METHOD

SPM is a baseline reference method for labeling affordances using feature vectors extracted from the previously mentioned pre-trained networks. This method works under the assumption that feature vectors corresponding to the same affordance lie within a shared affine space. To show this assumption, subspaces for affordance groups should be found first. The feature vectors of the learning part of the dataset are used to extract basis vectors for each affordance group. Figure 5 shows the general representation of SPM. Subspaces A and B are subspace representations of blue and purple data points, respectively. After calculating the subspaces A and B from the given blue and purple data points, the vector of new data shown by the red point is projected onto these subspaces. If the norm of projection is close to the norm of the original vector, then the new data point is labeled with the corresponding subspace label. In this context, feature vectors of the dataset images are labeled according to their ground truth affordance. For each affordance group i, feature vectors labeled with affordance i from the learning set are stacked into a matrix, denoted as $\mathbf{M}_i$. Then, the obtained matrix is subjected to Singular Value Decomposition (SVD) to get basis vectors of the affordance group i as shown in the equations (1) and (2).

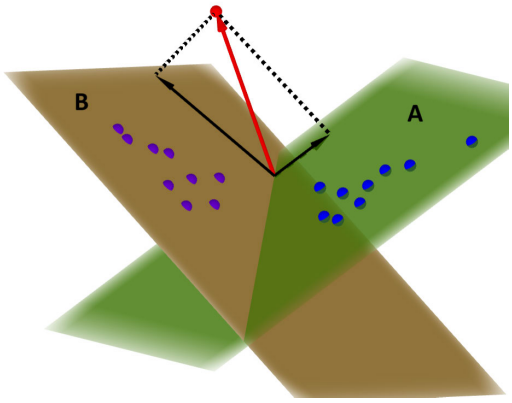

FIGURE 5. Projection of a feature vector onto two different subspaces belonging to two different affordances.

$$\mathbf{M_i} = horizontalStack(\mathbf{C_{i,learn}}) \tag{1}$$

$$\mathbf{U_i}\Sigma_\mathbf{i}\mathbf{V_i^T} = \mathbf{M_i} \tag{2}$$

SVD allows computing basis vectors of the subspace since columns of $\mathbf{U}_i$ are basis vectors for the feature vectors used to construct the $\mathbf{M}_i$ matrix. From the $\mathbf{U}_i$ matrix, the most significant d number of columns are selected as the basis vectors for the corresponding affordance space as $\mathbf{U}_i = \begin{bmatrix}\mathbf{u}_1 \ \mathbf{u}_2 \ \dots \ \mathbf{u}_d\end{bmatrix}$. Then, the projection matrix $\mathbf{P}_i$ for d dimensional subspace of the affordance group i is given simply in equation (3) since columns of $\mathbf{U}_i$ are already orthonormal. The calculation process of the projection matrices is shown in the algorithm 2.

$$\mathbf{P}_i = \mathbf{U}_i\mathbf{U}_i^T \tag{3}$$

The image's feature vector is projected onto each of the calculated subspaces to determine the appropriate affordance label(s) for an image from the validation set. Then, the ratio of the l_2 norm of the projected vector to that of the original vector is computed. Let $\mathbf{j}$ be a feature vector of an image

Algorithm 2 Projection Matrix Calculation

Input: $C_{learning}$: List of feature vectors of the learning set of all affordance groups

Output: P: List of projection matrices for all affordance groups

```
Initialize P
for C_{i,learning} in C_learning with affordance group i do
    M_i ← horizontalStack(C_{i,learning})
    U_i, Σ_i, V_i^T ← SVD(M_i)
    P_i ← U_i U_i^T
    Append P_i to P
end for
return P
```

from the validation set of the dataset, $C_{validation}$. Then, the l_2 norm ratio of the projected feature vector of the $\mathbf{j}$ onto d dimensional subspace of the affordance group $\mathbf{M}_i$ to original vector $\mathbf{j}$ shown in the equation (4).

$$d_{i,j} = \frac{|\mathbf{P}_i\,\mathbf{j}|_2}{|\mathbf{j}|_2} \tag{4}$$

If this ratio surpasses the threshold value, the corresponding affordance label is assigned to the image. A systematic process is followed to ascertain an appropriate threshold value, as shown in the Algorithm 3. Firstly, for each affordance group, all instances of the learning part of the feature vectors, $C_{learning}$, are investigated. For each affordance i, projection ratios of the feature vectors labeled with affordance i and non-labeled feature vectors with the affordance i are stored. Then, using True Positive (TP), False Negative (FN), False Positive (FP), and True Negative (TN) values, True Positive Rate (TPR) and False Positive Rate (FPR) are calculated for each threshold value ranging from 0 to 1. TPR and FPR calculations are shown in (5) and (6), respectively. The resulting TPR vs FPR values against increasing threshold values are given in Figure 6.

$$TPR = \frac{TP}{TP + FN} \tag{5}$$

$$FPR = \frac{FP}{FP + TN} \tag{6}$$

Computation of these metrics enables the determination of a threshold value that maximizes labeling performance while minimizing false negative labeling. After getting the TPR and FPR values for each threshold value for each affordance group, the threshold score denoted by ts for threshold values ranging from 0 to 1 for each affordance group separately are determined according to equation (7):

$$ts = \sqrt{(1 - TPR)^2 + FPR^2} \tag{7}$$

The presented ts value in equation (7) shows how close a particular point is to the $TPR = 1$, $FPR = 0$ point, representing 100% labeling performance. Hence, this closeness value, ts, is a labeling performance metric. After calculating ts values for each threshold value, the threshold value with the minimum ts is chosen as the threshold value for the affordance group. Figure 6 shows the optimal threshold TPR and FPR values for each affordance group with red dots on the graphs. As a result, the threshold values (shown as red dots in Figure 6) are selected for each affordance group. The threshold values for each affordance group are calculated in algorithm 3. Up to this point, only the learning subset of the dataset has been employed to compute the basis vectors and threshold values for each distinct affordance group. The next phase involves labeling the validation data. The process of the affordance labeling of a given image by SPM is given by the algorithm 4.

Algorithm 3 Threshold Value Determination

Input: $C_{learning}$: List of feature vectors of the learning set of affordance groups

P: List of projection matrices of affordance groups

Output: th: List of threshold values for each affordance group.

```
Initialize th
for i: affordance group in affordances do
    Initialize l_i: list of labeled projection ratios
    Initialize nl_i: list of unlabeled projection ratios
    for j: feature vector in C_learning do
        d_{i,j} ← norm(P_i j)/norm(j)
        if j has label i then
            Append d_{i,j} to l_i
        else
            Append d_{i,j} to nl_i
        end if
    end for
    Initialize th_i
    Initialize ts_i
    for thresh in range(0, 1) do
        Calculate ts_i
        if ts_i < previous ts_i then
            th_i = thresh
        end if
    end for
    Append th_i to th
end for
return th
```

As previously indicated, a subset of 13605 RGB images from the Large Scale Multi-View RGBD Affordance Learning Dataset and 18000 images from the RGB-D Part Affordance Dataset have been allocated for validation. The performance of the SPM is computed by calculating TPR and FPR values for each validation scene. Then, overall TPR and FPR values are used to evaluate labeling performance.

E. MANIFOLD CURVATURE METHOD

In this approach, similar to the SPM, the initial step involves constructing affordance label clusters for the feature vectors of the images. The learning phase only groups these vectors based on their respective affordance labels. When a new

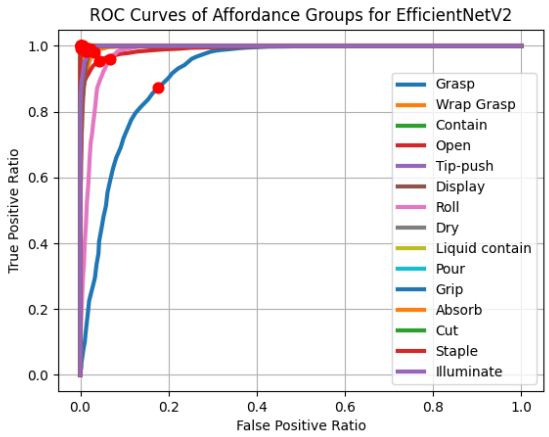

FIGURE 6. ROC curves of pre-trained EfficientNetV2 output of learning set of "A large scale multi-view RBD visual affordance learning dataset [43]".

Algorithm 4 Affordance Labeling via SPM

Input: $C_{validation}$: List of feature vectors of the validation set of affordances,
th: List of threshold values for Affordance groups,
P: List of affordance subspace projection matrices
Output: *aff*: List of indices of objects labeled with corresponding affordance.

```
Initialize aff
M_validation ← horizontalStack(C_validation)
for P_i:Projection matrix, th_i:threshold value of affordance
    i in P and th do
    projections ← P_i M_validation
     Append args(projections > th_i) to aff
end for
return aff
```

image is processed and a feature vector is generated using a pre-trained feature extractor, manifold curvature calculation is carried out to perform affordance labeling. Figure 7 shows two different cases of the MCM classification. In both cases, neighboring points, shown by blue dots, of the affordance manifold, shown by grey surface, are to be tested. The test points on both occasions are shown in red. The local subspace of the neighboring points is shown by a green plane. When the test point is added to this neighborhood, the resulting subspace of the new neighborhood is shown by a red plane. MCM aims to calculate weighted angles between the red and green planes, or subspaces in general, to understand how much a test point is suited to the local neighborhood. A small weighted angle will result in a better suit for the local neighborhood. In this case, the test point on Figure 7(a) results in a smaller angular change from the subspace of the local neighborhood. Hence, it can be concluded that the test point in Figure 7(a) is better suited to this manifold than Figure 7(b). Let $\mathbf{j}$ denote the feature vector of a test image from $C_{validation}$, and let C_i represent the previously established affordance label cluster for affordance i. Initially, n vectors within C_i with the smallest l_2 distances to vector $\mathbf{j}$ are selected. These vectors are denoted as $\mathbf{p}_{i,1}, \mathbf{p}_{i,2}, \ldots, \mathbf{p}_{i,n}$.

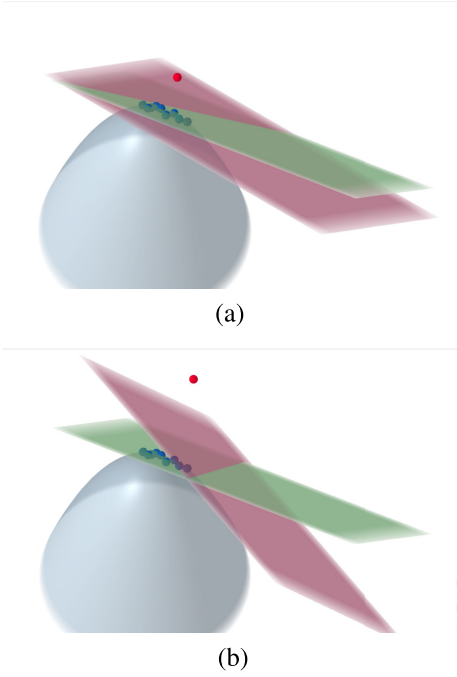

FIGURE 7. Graphical representation of MCM with different test points on a manifold.

Once the closest n points from C_i are found, two matrices are formed: $\tilde{\mathbf{M}}_{i,j}$, which consists of the local neighborhood vectors, and $\mathbf{M}_{i,j}$, which includes the feature vector $\mathbf{j}$ alongside the local neighborhood vectors. $\tilde{\mathbf{M}}_{i,j}$ and $\mathbf{M}_{i,j}$ are formed as follows:

$$\tilde{\mathbf{M}}_{i,j} = \begin{bmatrix} \mathbf{p}_{i,1} & \mathbf{p}_{i,2} & \cdots & \mathbf{p}_{i,n} \end{bmatrix}$$
$$\mathbf{M}_{i,j} = \begin{bmatrix} j & \tilde{\mathbf{M}}_{i,j} \end{bmatrix} = \begin{bmatrix} \mathbf{j} & \mathbf{p}_{i,1} & \mathbf{p}_{i,2} & \cdots & \mathbf{p}_{i,n} \end{bmatrix}$$

The objective here is to understand the local effect of the new data point $\mathbf{j}$ when it is introduced into a manifold. This is achieved by comparing matrices $\tilde{\mathbf{M}}_{i,j}$ and $\mathbf{M}_{i,j}$. Following the construction of these matrices, the skinny SVD is computed for each of them, where the skinny SVD only uses the columns of $\mathbf{U}$ and rows of $\mathbf{V}^T$ that correspond to non-zero singular values. Equations (8) and (9) are based on skinny SVD:

$$\mathbf{U}_{i,j}\Sigma_{i,j}\mathbf{V}_{i,j}^T = \mathbf{M}_{i,j} \tag{8}$$
$$\tilde{\mathbf{U}}_{i,j}\tilde{\Sigma}_{i,j}\tilde{\mathbf{V}}_{i,j}^T = \tilde{\mathbf{M}}_{i,j} \tag{9}$$

After obtaining these decompositions, how the local subspace changes when the vector $\mathbf{j}$ is included or excluded from the neighborhood can be determined. Since the columns of the $\mathbf{U}$ matrix represent unit basis vectors of the neighborhood, the $\mathbf{U}$ matrices can be utilized to find the principal angles between two subspaces. Based on $\mathbf{U}_{i,j}$ and $\tilde{\mathbf{U}}_{i,j}$, matrix $\mathbf{R}_{i,j}$ is defined

in equation (10):

$$\mathbf{R}_{i,j} = \mathbf{U}_{i,j}^T \tilde{\mathbf{U}}_{i,j} \tag{10}$$

In this context, the diagonal elements of matrix $\mathbf{R}_{i,j}$ are $\cos\theta_{1,1}, \cos\theta_{2,2}, \ldots, \cos\theta_{n,n}$, where $\theta_{n,n}$ represents the angle between the n^{th} columns of $\mathbf{U}_{i,j}$ and $\tilde{\mathbf{U}}_{i,j}$. The angle θ is calculated simply by summing the diagonal entries as shown in equation (11) to indicate some cumulative agreement between 2 subspaces as shown in equation (11):

$$\theta = \arccos \sum_{k=0}^{n} r_{k,k} \tag{11}$$

However, note that the order of the columns in the $\mathbf{U}$ matrix is important, and their weight in calculating the original matrix is represented by Σ. Therefore, equation (11) can be refined to reflect the importance of agreement/disagreement between dominant (i.e., principal) directions to result in yet a more precise angle calculation [60]. Thus, $\mathbf{R}_{i,j}$ matrix is redefined as shown in equation (12):

$$\mathbf{R}_{i,j} = (\mathbf{U}_{i,j}\Sigma_{i,j})^T (\tilde{\mathbf{U}}_{i,j}\tilde{\Sigma}_{i,j}) \tag{12}$$

Then, $\mathbf{R}_{i,j}$ matrix is decomposed using SVD in equation (13).

$$\mathbf{U}''_{i,j}\Sigma''_{i,j}\mathbf{V}''^T_{i,j} = \mathbf{R}_{i,j} \tag{13}$$

Lastly, an expression similar to equation (11) is calculated as shown in the equation (14) for the updated $\mathbf{R}_{i,j}$ matrix:

$$\theta_w = \arccos \frac{\sum_{k=0}^{n} \sigma''_{i,j_{kk}}}{\sum_{k=0}^{n} \sigma_{i,j_{kk}} \tilde{\sigma}_{i,j_{kk}}} \tag{14}$$

where $\sigma''_{i,j_{kk}}$, $\sigma_{i,j_{kk}}$ and $\tilde{\sigma}_{i,j_{kk}}$ are the k^{th} diagonal entries of $\Sigma''_{i,j}$, $\Sigma_{i,j}$ and $\tilde{\Sigma}_{i,j}$ respectively. This angle-sum value indicates the alteration within the local subspace with the introduction of **j** to this local subspace. Here, smaller angle values signify less variation or distortion in the local subspace. Hence, it indicates that **j** is in harmony/agreement with that local subspace. In other words, if introducing the feature vector of a test image to the local neighborhood within an affordance set results in a minimal angle-sum value, it suggests that this new image can be affiliated with this affordance set. A threshold value for each affordance group is calculated similarly to the threshold calculation for SPM. Therefore, if the angle-sum between the local subspaces of $\mathbf{M}_{i,j}$ and $\tilde{\mathbf{M}}_{i,j}$ is smaller than the threshold value, it implies that the image represented by the feature vector **j** should be labeled with affordance label i. This label corresponds to the group comprising the pre-defined vectors from the neighborhood from affordance cluster i, namely $\mathbf{p}_{i,1}, \mathbf{p}_{i,2}, \ldots, \mathbf{p}_{i,n}$. The whole process of labeling via MCM is presented in Algorithm 5.

Algorithm 5 Affordance Labeling via MCM

Input: $C_{learning}$, $C_{validation}$:List of learning and validation feature vector sets,
n : number of neighbour vectors
threshold: Threshold value for local subspace angle change
Output: *aff*: List of indices of objects labeled with corresponding affordance.

```
Initialize aff
for C_{i,learning} in C_learning with affordance group i do
    Initialize aff_i
    for j in C_validation do
        C_neighbours ← n vectors closest to j in C_{i,learning}
        M̃_{i,j} ← horizontalStack(C_neighbours)
        M_{i,j} ← horizontalStack(C_neighbours, j)
        U_{i,j}, Σ_{i,j}, V^T_{i,j} ← SVD(M_{i,j})
        Ũ_{i,j}, Σ̃_{i,j}, Ṽ^T_{i,j} ← SVD(M̃_{i,j})
        R_{i,j} ← (U_{i,j}Σ_{i,j})^T (Ũ_{i,j}Σ̃_{i,j})
        U_{i,j}, Σ_{ij,R}, V^T_{ij,R} ← SVD(R_ij)
        diag_sum''_{i,j} ← sum(diag(Σ''_{i,j}))
        diag_sum_{i,j} ← sum(diag(Σ_{i,j}Σ̃_{i,j}))
        Calculate local subspace change angle as θ_{i,j} ←
            arccos(diag_sum''_{i,j}/diag_sum_{i,j})
        if θ_{i,j} ≤ threshold then
            Append Arg(j) to aff_i
        end if
    end for
    Append aff_i to aff
end for
return aff
```

III. RESULTS AND DISCUSSION

In this section, the results of the proposed methods are tabulated and discussed. In this context, a selection of pre-trained neural networks, namely, ResNet-18, ResNet-50, ResNet-101, ResNet-152, ResNext-101, RegNetY, EfficientNetV2, ViT-L/16, and ViT-B/16 have been used in performance testing of the proposed methods. These networks have been pre-trained on the ImageNet [56] dataset. Due to the distinct architectures of these networks, their respective feature vector dimensions also differ.

Given the scarcity of studies addressing affordance labeling without segmentation, comparisons with methodologies in multi-label classification are made to evaluate the efficacy of our proposed SPM and MCM.

A. PERFORMANCE ANALYSIS OF AFFORDANCE LABELING METHODS

The basis vectors, projection matrices, and threshold values corresponding to the affordance classes are computed for each pre-trained network. As performance evaluation metrics, TPR and FPR explained in Section II-D are selected since pixel-level annotation is not aimed, and multiple labeling can be done to an image. In addition to TPR and FPR, Intersection over Union (IoU) is also calculated to measure the affordance labeling outputs of the validation set. IoU calculation is given in the Equation (15):

$$IoU = \frac{TP}{TP + FN + FP} \tag{15}$$

Furthermore, Precision and Recall are also considered to evaluate classification performance. These metrics are particularly important in multi-label affordance classification, where each image can be associated with multiple affordance labels.

Precision measures the proportion of correctly predicted affordances among all predicted affordances and is calculated as shown in Equation (16):

$$Precision = \frac{TP}{TP + FP} \tag{16}$$

A high precision indicates that the method has a low FPR, meaning it assigns affordance labels correctly with minimal misclassification. Recall measures the proportion of correctly predicted affordances among all actual affordances and is calculated as shown in Equation (17):

$$Recall = \frac{TP}{TP + FN} \tag{17}$$

A high recall indicates that most of the actual affordances have been detected, with fewer FNs.

To provide a more comprehensive evaluation, mAP is included as an additional performance measure. mAP metric computes the area under the Precision-Recall curve for each affordance class and then averages these values. This metric effectively balances precision and recall, ensuring a robust assessment of affordance labeling performance.

Using the given performance metrics, Table 1 and Table 2 give the affordance labeling performance results of SPM and MCM using various pre-trained network outputs as feature vectors for two different datasets.

Our analysis reveals that even the base method for comparison, SPM, which relies on the assumption that data coming from the same labels are distributed on or around some linear subspaces, resulted in performance comparable to some of the results in the literature or gives better results. It should be noted that variation from one network to another SPM performance changes more compared to MCM. This can be seen as the ability of the tested feature extractor pre-trained network's ability to flatten input data coming from the same cluster as it progresses toward the end of the network. Therefore, in image-based tasks where some subspace analysis tools are to be applied without additional training, EfficientNetV2, and ViT-L/16 are expected to give slightly better results compared to other tested pre-trained networks. Moreover, since the feature vector dimension of tested pre-trained ViT is lower than the other tested networks except for ResNet18, they will give lower computation times due to this feature vector dimension.

MCM yielded results on par with the current state of the art in affordance labeling. Comparatively, it is seen that MCM surpasses SPM by delivering consistently better results across various feature extractor networks, which makes it a consistent method for different architectures. Notably, MCM also yields lower False Positive labeling rates. These findings suggest that while pre-trained networks trained on the ImageNet [56] dataset can be used to assess subspaces of different classes decoded by one-hot encoding, dividing the outputs into multiple subspaces and evaluating for multi-hot encoded cases introduces complexities due to subspace intersections. Additionally, SPM is heavily reliant on the pre-trained network's capacity to group similar features into the same subspaces. In contrast, MCM excels because it evaluates data based on its appendability to existing data clusters, making it more transferable across different networks. Furthermore, by assessing the test vector's appendability to class clusters, this method offers flexibility for assigning multiple labels in intersection regions or no label at all.

Khalifa and Shah [43] focused on the potential of the Large Scale Affordance Dataset utilized in this study by conducting experiments with various segmentation and labeling networks. Their efforts resulted in a mean accuracy of 63.38% for the affordance segmentation task. Additionally, they explored transfer learning applications by adapting the last layer of pre-trained networks and training them to label affordances within the dataset they presented, achieving a mean accuracy of 91.83%. Furthermore, they introduced a novel Visual Affordance Transformer [61] in a subsequent study, which led to updated affordance segmentation results. The study used IOU as the performance metric, with results provided for each affordance category. The highest IOU accuracy obtained was 85.65% for the 'grasp' affordance, while the lowest was 41.85% for the 'cut' affordance. Since this study focuses on labeling without segmentation, comparing labeling performances is feasible.

SPM's highest labeling accuracy, with the highest TPR reaching 94.18% and the highest IoU 82.72% with ViT-L/16 [59] for A large scale multi-view RGBD visual affordance learning dataset, surpassing Khalifa and Shah's [43] 91.83% labeling accuracy. In addition, SPM achieves 96.73% TPR and 81.56% IoU accuracy over RGB-D Part Affordance Dataset with EfficientNetV2 [58] as the pre-trained network. Moreover, MCM consistently yields better labeling results across various pre-trained neural networks, achieving a TPR of 98.04% and IoU of 92.89% with ResNet-101 [3] and RegNet-Y [57] over A large scale multi-view RGBD visual affordance learning dataset, exceeding that of the SPM. Moreover, MCM achieves over 99% TPR and 97% IoU accuracy over RGB-D Part affordance Dataset using EfficientNetV2 [58] as the pre-trained network. Thus, MCM consistently outperforms the SPM in terms of TPR, FPR, and IoU labeling outcomes. Table 3 showcases the performance of MCM and SPM in affordance labeling. While both approaches accurately predict some ground truth affordances (e.g., rollability and wrap-graspability of glue sticks, and bottles), there are occasional errors (e.g., labeling illumination affordance for a ball).

Interestingly, these methods also identify affordances not explicitly labeled in the ground truth but demonstrably applicable to the object. For instance, SPM correctly assigns the 'tip-push' affordance to the cell phone, even though the

TABLE 1. Affordance labeling performances on a large scale multi-view RGBD visual affordance learning dataset.

Network Name	Feature Vector Size	SPM				MCM			
		TPR(%)	FPR(%)	IoU(%)	mAP(%)	TPR(%)	FPR(%)	IoU(%)	mAP(%)
ResNet-18	512	89.65	5.43	70.93	89.51	97.45	1.66	90.57	95.57
ResNet-50	2048	91.20	3.63	76.96	92.49	97.74	1.44	91.52	96.06
ResNet-101	2048	91.76	3.74	77.37	93.00	**98.04**	1.48	91.53	96.00
ResNet-152	2048	90.40	3.66	76.44	92.02	98.02	1.20	92.50	**96.58**
RegNet-Y	3024	91.07	3.09	78.50	94.25	97.15	0.90	**92.89**	96.49
EfficientNetV2	2560	92.07	**2.27**	82.20	**95.24**	93.95	0.88	89.58	95.63
ResNext101	2048	90.92	2.83	79.39	92.15	96.27	1.02	91.45	96.01
ViT-L/16	1024	**94.18**	2.64	**82.72**	94.76	93.73	**0.84**	89.44	95.42
ViT-B/16	768	91.86	2.28	81.94	94.66	97.23	0.98	92.54	96.54

TABLE 2. Affordance labeling performances on RGB-D Part affordance dataset.

Network Name	Feature Vector Size	SPM				MCM			
		TPR(%)	FPR(%)	IoU(%)	mAP(%)	TPR(%)	FPR(%)	IoU(%)	mAP(%)
ResNet-18	512	87.88	19.81	61.76	82.58	99.17	3.38	93.59	99.38
ResNet-50	2048	90.84	20.55	63.62	85.94	99.07	1.74	96.01	99.68
ResNet-101	2048	93.97	16.51	70.55	88.66	98.97	1.46	96.37	99.65
ResNet-152	2048	93.69	14.37	72.64	91.83	99.38	1.27	97.06	99.70
RegNet-Y	3024	93.90	11.44	76.36	95.53	98.99	**0.58**	97.90	99.78
EfficientNetV2	2560	**96.73**	**9.57**	**81.56**	**96.00**	**99.58**	0.91	**97.93**	**99.81**
ResNext101	2048	93.27	14.27	74.40	92.15	98.68	0.96	96.97	99.72
ViT-L/16	1024	94.08	10.29	77.69	94.05	98.29	0.79	96.95	99.60
ViT-B/16	768	95.88	9.89	79.75	93.83	99.17	0.81	97.69	99.76

keyboard is not visible. Both methods assign 'roll' affordance to food can as can be seen from Table 3, which are not present in the ground truth but are valid based on the object's physical properties. This suggests the model recognizes a button-like element based on its location. The findings from Table 3 suggest that both MCM and SPM demonstrate promising performance in affordance labeling. Notably, their ability to capture affordances not explicitly labeled in the ground truth makes them valuable tools for exploring new potential interactions with objects. These methods can potentially reveal previously unconsidered affordances, leading to a broader planning space for robots and enhancing robot-object-human interaction.

B. ERROR ANALYSIS

An analysis of affordance labeling errors, such as incorrectly assigning the illumination affordance to a ball object should be done to understand the causes of errors for the methods. Both SPM and MCM depend on the feature vectors of the learning set. Since one object may afford multiple actions, the feature vector of this object may be in various affordance groups. Hence, affordance groups have intersection regions at those points. In other words, if it is assumed that feature vectors with the same affordance label lie on the same manifold, then it can be said that these manifolds overlap by large since many objects afford several if not many labels. Thus, affordance groups are interlaced at feature vectors of the objects that afford various affordances. For example, if an object affords 'A' and 'B' affordance, manifolds of affordance groups 'A' and 'B' intersect at this point. Since multiple objects have multiple affordances, affordance group manifolds are interlaced with one another at multiple points. Hence, there may be mislabeling of affordances where true labeling has a common close affordance group. For instance, in the 'ball' object example, labeled with 'wrap-grasp', 'roll', and 'grasp' affordances. There is a possibility of mislabeling 'tip-push' and 'illuminate' affordances to the ball object due to the 'torch' objects. Since there are numerous torch objects which afford 'grasp', 'wrap-grasp', 'roll', 'tip-push', and 'illuminate', manifolds of those affordance groups interlace at multiple points. This means that around those points, they are also close to each other. Due to this reason, if an object affords more affordances to this neighborhood, the likelihood of labeling it with the other affordance labels in this neighborhood increases.

C. COMPARATIVE ANALYSIS WITH EXISTING METHODS

In comparing the results of this study to existing approaches, while Khalifa and Shah [43] achieved a mean accuracy of 91.83% using modified networks, the best-performing proposed method (MCM with RegNet-Y) surpassed this with a TPR of 97.15%, IoU accuracy of 92.89% and mAP of 96.49% over the same dataset. This comparison underscores the effectiveness of the MCM, and even the benchmark method of this study, SPM, particularly in leveraging pre-trained networks without necessitating additional training.

Moreover, SPM and MCM exhibit the ability to identify affordances not annotated in the ground truth, highlighting their potential to enhance affordance exploration capabilities. This feature not only demonstrates the value of this study but also signifies its applicability across various robotic tasks requiring interaction with diverse objects.

TABLE 3. Labeling examples of validation set via SPM and MCM.

Object Image	Ground Truth	Labeling Method	Correct Labeled Affordances Marked in Ground Truth	Correct Labeled Affordances not Present in Ground Truth	False Labeled Affordances
	• Grasp	SPM		• Tip-Push • Display	
		MCM	• Grasp		
	• Wrap-Grasp • Open	SPM	• Wrap-Grasp • Open	• Roll	
		MCM	• Wrap-Grasp • Open		
	• Grasp • Tip-Push	SPM	• Grasp • Tip-Push	• Display	
		MCM	• Grasp • Tip-Push		
	• Grasp • Wrap-Grasp • Liquid Contain	SPM	• Grasp • Wrap-Grasp • Liquid Contain	• Pour	
		MCM	• Grasp • Wrap-Grasp • Liquid Contain	• Pour	
	• Wrap-Grasp • Open	SPM	• Wrap-Grasp • Open	• Roll	
		MCM	• Wrap-Grasp • Open	• Roll	
	• Grasp • Wrap-Grasp • Liquid Contain	SPM	• Wrap-Grasp • Liquid Contain	• Pour	
		MCM	• Grasp • Wrap-Grasp • Liquid Contain	• Pour	
	• Grasp	SPM	• Graspl	• Open	
		MCM	• Grasp		

D. COMPARISON WITH EXISTING MULTI-LABEL CLASSIFICATION METHODS

The primary objective of this study is to label affordances in an image, where multiple affordance labels can be assigned to the same image. In this regard, the proposed approach shares similarities with multi-label classification methods. Although conventional multi-label classification primarily focuses on detecting multiple distinct objects in an image,

the methods presented in this study focus on identifying the functional affordances of a single object. Due to the limited availability of affordance labeling studies that do not involve segmentation, we compare our results with state-of-the-art multi-label classification approaches, despite their differences in scope.

A key distinction of this study is that, unlike conventional multi-label classification, the same set of pixels in an image can be assigned multiple affordance labels. In contrast, multi-label classification typically assigns different labels to distinct regions of an image. Additionally, a fundamental challenge of this study lies in extracting affordance labels directly from pre-trained networks without any additional training. This is in contrast to traditional multi-label classification methods, which often involve fine-tuning networks specifically for the classification task.

Xu et al. [49] achieved state-of-the-art performance in multi-label classification on the MS COCO dataset, reporting a mean Average Precision (mAP) of 93.54% for images with a resolution of 1344 × 1344 pixels. When evaluated with images at a lower resolution of 224×224 pixels, their method exhibited a decrease in mAP, achieving 89.82%.

Similarly, Liu et al. [50] obtained remarkable results on the PASCAL VOC dataset, reaching an mAP of 96.6% for multi-label classification. Their model was trained using input images with a resolution of 448×448 pixels, containing four times more pixels than the 224 × 224 resolution utilized in this study.

Despite operating with lower-resolution images, the proposed methods achieved comparable or better results in terms of mAP. Specifically, SPM achieved an mAP exceeding 95%, while MCM exceeded 96% across both datasets. This highlights the robustness and effectiveness of the proposed methods, even under resource-constrained conditions. Furthermore, the training-free nature of SPM and MCM offers significant computational advantages, making them highly scalable and efficient for large-scale applications.

These results show the potential of the proposed approach in real-world scenarios, such as robotic perception and affordance-based object interaction. By demonstrating competitive performance with state-of-the-art multi-label classification methods while requiring no additional training, SPM and MCM present a promising alternative for affordance labeling tasks.

E. EXPERIMENTAL PARAMETERS AND THRESHOLD SELECTION

The selection of parameters plays a crucial role in the performance of the presented methods. In particular, the threshold values are integral to accurately labeling affordances. The systematic approach presented for threshold determination by calculating the TPR and FPR values across a range of threshold values ensures that labeling performance is optimized. Understanding the impact of threshold adjustments on labeling performance allows us to fine-tune the presented methods for enhanced accuracy.

IV. CONCLUSION

In this paper, two methods for identifying affordance labels of objects are introduced. These methods generate decisions by analyzing feature vectors extracted from existing pre-trained networks using subspace projections and so-called manifold curvatures, respectively. It must be noted that neither of these proposed methods requires any further training, which separates them from the ones existing in the literature. These methods have been evaluated using nine well-known pre-trained networks. Results indicate the efficacy of both the SPM and the MCM. SPM achieves TPR and IoU accuracies exceeding 94% and 82% over A large scale multi-view RGBD visual affordance learning dataset and over 96% and 81% over RGB-D Part affordance dataset. MCM achieves TPR and IoU accuracies exceeding 98% and 92% over A large scale multi-view RGBD visual affordance learning dataset and over 99% and 97% over RGB-D Part affordance dataset. This demonstrates the practical applicability of these approaches in affordance labeling.

Furthermore, observations of the labeling results suggest that the proposed methods can discover affordances of objects not explicitly labeled in the ground truth information. This is particularly evident in identifying the 'roll' affordance for cylindrical objects like cans, which are typically not labeled as 'roll' in the ground truth dataset. Conversely, the detection of 'open' affordance for toothpaste may not be as readily apparent. Such findings show the potential of the presented methods for affordance exploration. Additionally, owing to their reliance on feature vectors and vector operations, these methods offer flexibility in accommodating new affordance categories and incorporating new data into existing affordance groups easily, i.e., without the need for any training.

ACKNOWLEDGMENT

The authors thank the Center for Robotics and Artificial Intelligence (ROMER), Middle East Technical University, for its continuous support to this research.

REFERENCES

[1] J. Gibson, *The Senses Considered as Perceptual Systems*. London, U.K.: Bloomsbury Academic, 1966.
[2] J. Gibson, *The Ecological Approach to Visual Perception*. Boston, MA, USA: Houghton Mifflin, 1979.
[3] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," 2015, *arXiv:1512.03385*.
[4] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," 2016, *arXiv:1611.05431*.
[5] E. Ugur, Y. Nagai, E. Sahin, and E. Oztop, "Staged development of robot skills: Behavior formation, affordance learning and imitation with motionese," *IEEE Trans. Auto. Mental Develop.*, vol. 7, no. 2, pp. 119–139, Jun. 2015.
[6] J. Modayil and B. Kuipers, "The initial development of object knowledge by a learning robot," *Robot. Auto. Syst.*, vol. 56, no. 11, pp. 879–890, Nov. 2008.
[7] H. S. Koppula, R. Gupta, and A. Saxena, "Learning human activities and object affordances from RGB-D videos," *Int. J. Robot. Res.*, vol. 32, no. 8, pp. 951–970, Jul. 2013.
[8] J. Sawatzky, A. Srikantha, and J. Gall, "Weakly supervised affordance detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5197–5206.

[9] S. Thermos, P. Daras, and G. Potamianos, "A deep learning approach to object affordance segmentation," in *Proc. ICASSP - IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2020, pp. 2358–2362.

[10] A. Nguyen, D. Kanoulas, D. G. Caldwell, and N. G. Tsagarakis, "Detecting object affordances with convolutional neural networks," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2016, pp. 2765–2770.

[11] A. Iriondo, E. Lazkano, and A. Ansuategi, "Affordance-based grasping point detection using graph convolutional networks for industrial bin-picking applications," *Sensors*, vol. 21, no. 3, p. 816, Jan. 2021.

[12] H. Wu and G. S. Chirikjian, "Can i pour into it? Robot imagining open containability affordance of previously unseen objects via physical simulations," *IEEE Robot. Autom. Lett.*, vol. 6, no. 1, pp. 271–278, Jan. 2021.

[13] A. Myers, C. L. Teo, C. Fermüller, and Y. Aloimonos, "Affordance detection of tool parts from geometric features," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2015, pp. 1374–1381.

[14] S. Thermos, G. Potamianos, and P. Daras, "Joint object affordance reasoning and segmentation in RGB-D videos," *IEEE Access*, vol. 9, pp. 89699–89713, 2021.

[15] G. Li, V. Jampani, D. Sun, and L. Sevilla-Lara, "LOCATE: Localize and transfer object parts for weakly supervised affordance grounding," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Los Alamitos, CA, USA, Jun. 2023, pp. 10922–10931.

[16] T. V. Vo, M. N. Vu, B. Huang, T. Nguyen, N. Le, T. Vo, and A. Nguyen, "Open-vocabulary affordance detection using knowledge distillation and text-point correlation," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Sep. 2022, pp. 13508–13517.

[17] L. Mur-Labadia, J. J. Guerrero, and R. Martinez-Cantin, "Multi-label affordance mapping from egocentric vision," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2023, pp. 5215–5226.

[18] Y. Yang, W. Zhai, H. Luo, Y. Cao, and Z.-J. Zha, "LEMON: Learning 3D human-object interaction relation from 2D images," 2023, *arXiv:2312.08963*.

[19] H. Kim, S. Han, P. Kwon, and H. Joo, "Beyond the contact: Discovering comprehensive affordance for 3D objects from pre-trained 2D diffusion models," in *Computer Vision—ECCV*, A. Leonardis, E. Ricci, S. Roth, O. Russakovsky, T. Sattler, and G. Varol, Eds., Cham, Switzerland: Springer, 2025, pp. 400–419.

[20] E. Ragusa, S. Dosen, R. Zunino, and P. Gastaldo, "Affordance segmentation using tiny networks for sensing systems in wearable robotic devices," *IEEE Sensors J.*, vol. 23, no. 19, pp. 23916–23926, Oct. 2023.

[21] P. Ardón, M. E. Cabrera, È. Pairet, R. P. A. Petrick, S. Ramamoorthy, K. S. Lohan, and M. Cakmak, "Affordance-aware handovers with human arm mobility constraints," *IEEE Robot. Autom. Lett.*, vol. 6, no. 2, pp. 3136–3143, Apr. 2021.

[22] C. Chen, A. Seff, A. Kornhauser, and J. Xiao, "DeepDriving: Learning affordance for direct perception in autonomous driving," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 2722–2730.

[23] K. Ransikarbum, N. Kim, S. Ha, R. A. Wysk, and L. Rothrock, "A highway-driving system design viewpoint using an agent-based modeling of an affordance-based finite state automata," *IEEE Access*, vol. 6, pp. 2193–2205, 2018.

[24] S. Bozeat, M. A. L. Ralph, K. Patterson, and J. R. Hodges, "When objects lose their meaning: What happens to their use?" *Cognit., Affect., Behav. Neurosci.*, vol. 2, no. 3, pp. 236–251, Sep. 2002.

[25] G. Federico and M. A. Brandimonte, "Looking to recognise: The pre-eminence of semantic over sensorimotor processing in human tool use," *Sci. Rep.*, vol. 10, p. 6157, Dec. 2020.

[26] E. Şahin, M. Çakmak, M. R. Doğar, E. Uğur, and G. Üçoluk, "To afford or not to afford: A new formalization of affordances toward affordance-based robot control," *Adapt. Behav.*, vol. 15, no. 4, pp. 447–472, Dec. 2007.

[27] M. Andries, A. Dehban, and J. Santos-Victor, "Automatic generation of object shapes with desired affordances using voxelgrid representation," *Frontiers Neurorobotics*, vol. 14, p. 22, May 2020.

[28] T. Apicella, A. Xompero, P. Gastaldo, and A. Cavallaro, "Segmenting object affordances: Reproducibility and sensitivity to scale," 2024, *arXiv:2409.01814*.

[29] W. Chen, H. Liang, Z. Chen, F. Sun, and J. Zhang, "Learning 6-DoF task-oriented grasp detection via implicit estimation and visual affordance," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2022, pp. 762–769.

[30] G. Li, D. Sun, L. Sevilla-Lara, and V. Jampani, "One-shot open affordance learning with foundation models," 2023, *arXiv:2311.17776*.

[31] E. Ruiz and W. Mayol-Cuevas, "Geometric affordance perception: Leveraging deep 3D saliency with the interaction tensor," *Frontiers Neurorobotics*, vol. 14, p. 45, Jul. 2020.

[32] B. Moldovan, P. Moreno, M. van Otterlo, J. Santos-Victor, and L. De Raedt, "Learning relational affordance models for robots in multi-object manipulation tasks," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2012, pp. 4373–4378.

[33] E. Uğur, E. Şahin, and E. Öztop, "Affordance learning from range data for multi-step planning," in *Proc. Int. Conf. Epigenetic Robot.*, Jan. 2009.

[34] E. Ugur, E. Oztop, and E. Sahin, "Goal emulation and planning in perceptual space using learned affordances," *Robot. Auto. Syst.*, vol. 59, nos. 7–8, pp. 580–595, Jul. 2011.

[35] E. Ugur, M. R. Dogar, M. Cakmak, and E. Sahin, "The learning and use of traversability affordance using range images on a mobile robot," in *Proc. IEEE Int. Conf. Robot. Autom.*, Apr. 2007, pp. 1721–1726.

[36] D. Xu, A. Mandlekar, R. Martín-Martín, Y. Zhu, S. Savarese, and L. Fei-Fei, "Deep affordance foresight: Planning through what can be done in the future," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2021, pp. 6206–6213.

[37] M. Hassanin, S. Khan, and M. Tahtali, "A new localization objective for accurate fine-grained affordance segmentation under high-scale variations," *IEEE Access*, vol. 8, pp. 28123–28132, 2020.

[38] A. K. Pandey and R. Alami, "Affordance graph: A framework to encode perspective taking and effort based affordances for day-to-day human–robot interaction," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Nov. 2013, pp. 2180–2187.

[39] E. Ragusa, C. Gianoglio, S. Dosen, and P. Gastaldo, "Hardware-aware affordance detection for application in portable embedded systems," *IEEE Access*, vol. 9, pp. 123178–123193, 2021.

[40] Y. Li, N. Zhao, J. Xiao, C. Feng, X. Wang, and T.-S. Chua, "LASO: Language-guided affordance segmentation on 3D object," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Los Alamitos, CA, USA, Jun. 2024, pp. 14251–14260.

[41] H. Luo, W. Zhai, J. Zhang, Y. Cao, and D. Tao, "Leverage interactive affinity for affordance learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 6809–6819.

[42] R. Xu, Y. Shen, X. Li, R. Wu, and H. Dong, "NaturalVLM: Leveraging fine-grained natural language for affordance-guided visual manipulation," 2024, *arXiv:2403.08355*.

[43] Z. Khalifa and S. A. A. Shah, "A large scale multi-view RGBD visual affordance learning dataset," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2023, pp. 1325–1329.

[44] C.-Y. Chuang, J. Li, A. Torralba, and S. Fidler, "Learning to act properly: Predicting and explaining affordances from images," 2017, *arXiv:1712.07576*.

[45] J. Wang, Y. Yang, J. Mao, Z. Huang, C. Huang, and W. Xu, "CNN-RNN: A unified framework for multi-label image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2285–2294.

[46] Y. Li, Y. Song, and J. Luo, "Improving pairwise ranking for multi-label image classification," 2017, *arXiv:1704.03135*.

[47] E. Schultheis, W. Kotłowski, M. Wydmuch, R. Babbar, S. Borman, and K. Dembczyński, "Consistent algorithms for multi-label classification with macro-at-*k* metrics," 2024, *arXiv:2401.16594*.

[48] K. Prokofiev and V. Sovrasov, "Combining metric learning and attention heads for accurate and efficient multilabel image classification," 2022, *arXiv:2209.06585*.

[49] S. Xu, Y. Li, J. Hsiao, C. Ho, and Z. Qi, "Open vocabulary multi-label classification with dual-modal decoder on aligned visual-textual features," 2023, *arXiv:2208.09562*.

[50] S. Liu, L. Zhang, X. Yang, H. Su, and J. Zhu, "Query2Label: A simple transformer way to multi-label classification," 2021, *arXiv:2107.10834*.

[51] X. Zhu, J. Liu, W. Liu, J. Ge, B. Liu, and J. Cao, "Scene-aware label graph learning for multi-label image classification," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2023, pp. 1473–1482.

[52] J. Jia, F. He, N. Gao, X. Chen, and K. Huang, "Learning disentangled label representations for multi-label classification," 2022, *arXiv:2212.01461*.

[53] A. B. Koku, A. Cakir, M. Parlaktuna, and A. Sekmen, "To train or not to train," in *Proc. IEEE 14th Int. Conf. Control Autom. (ICCA)*, Jun. 2018, pp. 835–840.

[54] A. Sekmen, M. Parlaktuna, A. Abdul-Malek, E. Erdemir, and A. B. Koku, "Robust feature space separation for deep convolutional neural network training," *Discover Artif. Intell.*, vol. 1, no. 1, pp. 1–15, Dec. 2021.

[55] A. Aldroubi, K. Hamm, A. B. Koku, and A. Sekmen, "CUR decompositions, similarity matrices, and subspace clustering," *Frontiers Appl. Math. Statist.*, vol. 4, p. 65, Jan. 2019.
[56] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. S. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Apr. 2015.
[57] I. Radosavovic, R. P. Kosaraju, R. Girshick, K. He, and P. Dollár, "Designing network design spaces," 2020, *arXiv:2003.13678*.
[58] M. Tan and Q. V. Le, "EfficientNetV2: Smaller models and faster training," 2021, *arXiv:2104.00298*.
[59] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16×16 words: Transformers for image recognition at scale," 2020, *arXiv:2010.11929*.
[60] A. Sekmen and B. Bilgin, "Manifold curvature estimation for neural networks," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Dec. 2022, pp. 3903–3908.
[61] S. A. A. Shah and Z. Khalifa, "Hierarchical transformer for visual affordance understanding using a large-scale dataset," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2023, pp. 11371–11376.

İSMAİL ÖZÇİL received the B.S., M.S., and Ph.D. degrees in mechanical engineering from Middle East Technical University, Ankara, Türkiye, in 2015, 2018, and 2025, respectively. He has been a Teaching Assistant with the Department of Mechanical Engineering, Middle East Technical University, since 2015. His research interests include robotics, mechatronics, and machine learning.

A. BUĞRA KOKU received the B.S. degree in mechanical engineering and the M.S. degree in systems and control engineering from Boğaziçi University, İstanbul Türkiye, in 1994 and 1997, respectively, and the Ph.D. degree in electrical engineering and computer science from Vanderbilt University, Nashville, TN, USA, in 2003. He has been with the Department of Mechanical Engineering, Middle East Technical University (METU), since 2003, and a Vice Chair of the Center for Robotics and Artificial Intelligence (ROMER (romer.metu.edu.tr)), Middle East Technical University, since 2021. His research interest includes robotics. He is currently conducting research on topics ranging from the design of mobile robots to control of outdoor mobile robots, from the design of human-like robots to human–robot interaction.